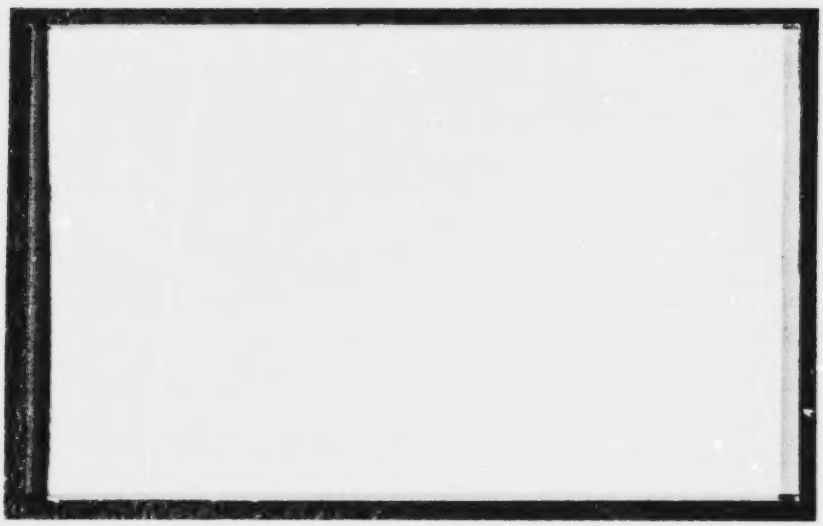
UNIVERSITÉ DE MONTRÉAL



CENTRE DE RECHERCHE SUR LES TRANSPORTS

MÉMOIRE D'ÉTUDIANT

CASE POSTALE 6128, SUCCURSALE A, MONTRÉAL, QUÉBEC, H3C 3J7

CONSTRUCTION D'ALGORITHMES OPTIMAUX POUR DES EXTENSIONS AU PROBLÈME DU VOYAGEUR DE COMMERCE

par

Yves Nobert

Thèse de Ph.D., département d'informatique et de recherche opérationnelle, dirigée par Gilbert Laporte

Université de Montréal

Centre de recherche sur les transports - Publication #297

Mai 1982

Dépôt légal - 2e trimestre 1983

TABLE DES MATIERES

Page

SOMMAIRE xi

CHAPITRE 1 : DESCRIPTION DE NOTRE RECHERCHE 1

1.1 - Quatre généralisations du problème du voyageur de commerce 2

1.2 - Principales étapes de notre recherche 5

1.2.1 - Génération des problèmes-tests aléatoires 6

1.3 - Notation et terminologie générales 7

1.4 - Terminologie de la théorie des graphes 7

1.4.1 - Terminologie associée aux graphes orientés 7

1.4.2 - Terminologie associée aux graphes non-orientés 12

1.4.3 - Correspondance entre un graphe et un problème du type PLE ou PEV 15

1.5 - Principales variables et notations utilisées dans les modèles mathématiques 16

1.6 - Outils informatiques 18

1.6.1 - Les programmes MIF et BB 18

1.6.2 - Principales modifications apportées à MIF et BB 24

CHAPITRE 2 : LE PROBLEME DU VOYAGEUR DE COMMERCE 28

2.1 - Introduction 29

2.2 - Formulations des PVC asymétriques 31

2.2.1 - Introduction 31

2.2.2 - Contraintes d'élimination de sous-tours de Dantzig et al. 33

2.2.3 - Contraintes d'élimination de sous-tours de Miller et al. 35

2.2.4 - Contraintes d'élimination de sous-tours de Gavish et Graves 37

2.3 - Algorithmes optimaux pour les PVC asymétriques 39

Page

2.3.1 - Introduction 39
2.3.2 - Méthodes de "Branch-and-Bound" pour le PVC asymétrique basées sur le problème d'affectation 40
2.3.3 - Bornes inférieures différentes de V(PA) sur V(PVC) 46
2.4 - Formulations des PVC symétriques 47
2.5 - Algorithmes optimaux pour les PVC symétriques 50
2.6 - Résolution des PVC symétriques par l'utilisation de la programmation linéaire en nombres entiers 53
2.6.1 - Introduction 53
2.6.2 - Résolution du modèle de Dantzig et al. 54
2.7 - Conclusion 62

CHAPITRE 3 : LE PROBLEME DES m VOYAGEURS DE COMMERCE (m-PVC) 63

3.1 - Introduction 64
3.2 - Transformation du m-PVC en un PVC 68
3.2.1 - Introduction 68
3.2.2 - La transformation d'Eilon et al. 71
3.2.3 - La transformation de Lenstra et Rinnoy Kan 72
3.2.4 - La transformation de Bellmore et Hong 74
3.2.5 - La transformation de Hong et Padberg 77
3.2.6 - La transformation de Rao 80
3.3 - Formulation des m-PVC asymétriques 80
3.3.1 - Introduction 80
3.3.2 - Contraintes d'élimination de sous-tours illégaux de Miller et al. 83
3.3.3 - Nos contraintes d'élimination de sous-tours illégaux ... 83
3.3.4 - Contraintes d'élimination de sous-tours illégaux de Gavish et Graves 84
3.3.5 - Contraintes d'élimination de sous-tours illégaux de Svestka et Huckfeldt 84
3.4 - Formulations des m-PVC symétriques 86

Page

3.4.1 - Introduction 86
3.4.2 - Notre modèle pour le m-PVC symétrique 86
3.4.3 - Le modèle de Gavish et Srikanth pour le m-PVC symétrique 88

3.5 - La construction d'algorithmes optimaux pour le m-PVC: commentaires généraux et revue de la littérature 90

3.6 - Deux nouveaux algorithmes pour le m-PVC 93
3.6.1 - Introduction 93
3.5.2 - Description générale de nos deux algorithmes 93
3.6.3 - Description technique de nos deux algorithmes 98
3.6.4 - Implantation sur ordinateur de NORMAL et RENVERSE 103
3.6.5 - Tests de performance 105

3.7 - Comparaison de la performance de nos deux algorithmes versus celle des autres méthodes résolvant le m-PVC 113

3.8 - Conclusion 115

CHAPITRE 4 : LE PROBLEME DES m VOYAGEURS DE COMMERCE ASSUJETTIS A UNE RESTRICTION DE CAPACITE (c.m-PVC) 117

4.1 - Introduction 118

4.2 - Définition du c.m-PVC 119

4.3 - Formulations mathématiques pour le c.m-PVC 121
4.3.1 - Le modèle de Balinski et Quandt 121
4.3.2 - Notre modèle pour le c.m-PVC symétrique 123
4.3.3 - Cas asymétrique 126

4.4 - Notre modèle et celui de Gavish et Graves pour le c.m-PVC asymétrique 129
4.4.1 - Introduction 129
4.4.2 - Description de notre modèle et de celui de Gavish et Graves pour le c.m-PVC asymétrique 130

Page

4.4.3 - Application de la technique de décomposition de Benders au modèle de Gavish et Graves 135

4.5 - Résumé de la littérature concernant les méthodes de résolution du c.m-PVC 144

4.5.1 - Méthodes approximatives pour le c.m-PVC 144

4.5.2 - Résumé des principales méthodes exactes de résolution du c.m-PVC 148

4.6 - Une nouvelle méthode optimale pour le c.m-PVC 150

4.6.1 - Description générale 150

4.6.2 - Implantation sur ordinateur 154

4.6.3 - Résultats de tests sur ordinateur 166

4.6.4 - Comparaison avec les autres méthodes pour le c.m-PVC ... 177

4.7 - Conclusion 178

CHAPITRE 5 : LE PROBLEME DES m VOYAGEURS DE COMMERCE AVEC p DEPOTS ((p,m)-PVC) 180

5.1 - Introduction 181

5.2 - Formulation du (p,m)-PVC 182

5.3 - Une méthode exacte de résolution du (p,m)-PVC 188

5.4 - Tests de performance 193

5.5 - Conclusion 194

CHAPITRE 6 : LE PROBLEME DES m VOYAGEURS DE COMMERCE AVEC DEPOT VARIABLE (v.m-PVC) 197

6.1 - Introduction 198

6.2 - Définition et formulation du v.m-PVC 201

6.2.1 - Définition et notation 201

6.2.2 - Formulation du v.m-PVC 202

Page

6.3 - Une méthode exacte de résolution du v.m-PVC 207

6.4 - Résultats sur ordinateur 214

6.5 - Conclusion 222

CHAPITRE 7 : CONCLUSIONS ET EXTENSIONS 224

7.1 - Conclusion 225

7.2 - Extensions 226

REMERCIEMENTS 229

BIBLIOGRAPHIE 230

*
* *

TABLE DES ILLUSTRATIONS

- FIGURES -

Page

CHAPITRE 1

Figure 1.1 : Exemple d'un graphe orienté G = (N,A) 11

Figure 1.2 : Exemple d'un graphe non-orienté 15

Figure 1.3 : Programme MIF 21

Figure 1.4 : Programme BB 24

Figure 1.5 : Modification type effectuée dans MIF 25

CHAPITRE 2

Figure 2.1 : Algorithmes de Miliotis: première variante 58

CHAPITRE 3

Figure 3.1 : Correspondance entre un 2-tour et un tour par la transformation d'Eilon et al. 72

Figure 3.2 : Illustration de la transformation de Lenstra et Rinnoy Kan 74

Figure 3.3 : Illustration de la transformation de Bellmore et Hong ... 76

Figure 3.4 : Illustration de la transformation de Hong et Padberg 79

Figure 3.5 : Exemple d'un sous-tour illégal contenant n-2 villes 88

Figure 3.6 : Algorithme RENVERSE 104

CHAPITRE 4

Figure 4.1 : Exemple d'une solution irréalisable pour le c.m-PVC 125

Figure 4.2 : Calcul des "savings" de Clarke et Wright 146

Page

Figure 4.3 : Une méthode de type RENVERSE n'est pas valide pour le c.m-PVC 153

Figure 4.4 : Organigramme de notre algorithme pour le c.m-PVC 155

Figure 4.5 : Modification de la structure de données si x_{1j} est fixée à 1 159

Figure 4.6 : Modification de la structure de données si x_{ij} est fixée à 1 et si $i \neq 1$ .....

Figure 4.7 : Exemple de la construction de R 164

CHAPITRE 5

Figure 5.1 : Illustration de la non validité de (5.4) pour les sous-tours de plus d'un dépôt 187

Figure 5.2 : Exemple d'une composante connexe dans une solution fractionnaire 190

Figure 5.3 : Exemple d'une composante connexe avec 4 dépôts 191

CHAPITRE 6

Figure 6.1 : Organigramme de notre algorithme pour le v.m-PVC 209

Figure 6.2 : Première stratégie de branchement 212

Figure 6.3 : Deuxième stratégie de branchement 213

* * *

TABLE DES ILLUSTRATIONS

- TABLEAUX -

Page

CHAPITRE 1

Tableau 1.1 : Description du PVC et de ses 4 généralisations considérées 5

CHAPITRE 3

Tableau 3.1 : Nombre de contraintes dans (3.38) selon leur forme et selon le type de C 91

Tableau 3.2 : Comparaison entre l'algorithme NORMAL et l'algorithme RENVERSE 106

Tableau 3.3 : Tests de performance sur le $\bar{m}$-PVC 111

Tableau 3.4 : Résolution d'un $\bar{m}$-PVC par transformation en un PVC(A) et résolution directe d'un $\bar{m}$-PVC par RENVERSE (B) 113

CHAPITRE 4

Tableau 4.1 : Les 8 versions de notre algorithme pour le c.m-PVC 168

Tableau 4.2 : Comparaison des 8 versions 169

Tableau 4.3 : Problèmes non-euclidiens, stratégie de branchement de Land et Powell 172

Tableau 4.4 : Problèmes non-euclidiens, notre stratégie de branchement 173

Tableau 4.5 : Problèmes euclidiens, stratégie de branchement de Land et Powell 174

Tableau 4.6 : Problèmes euclidiens, notre stratégie de branchement ... 175

Page

CHAPITRE 5

Tableau 5.1 : Problèmes non-euclidiens 195

Tableau 5.2 : Problèmes euclidiens 196

CHAPITRE 6

Tableau 6.1 : Résolution du v.m-PVC et n résolutions d'un m-PVC 215

Tableau 6.2 : Problèmes euclidiens, première stratégie de branchement. 217

Tableau 6.3 : Problèmes euclidiens, deuxième stratégie de branchement. 218

Tableau 6.4 : Problèmes non-euclidiens, première stratégie de branchement 219

Tableau 6.5 : Problèmes non-euclidiens, deuxième stratégie de branchement 220

Tableau 6.6 : Temps de résolution avec coûts f_k non-nuls sur 10 problèmes de 20 villes 221

*
* *

SOMMAIRE

Cette étude concerne la construction de méthodes exactes de résolution de quatre généralisations du problème du voyageur de commerce (PVC). Ce problème consiste à trouver la route de distance totale minimale pour un voyageur de commerce qui doit visiter un certain nombre de villes, chacune exactement une fois, puis revenir à son point de départ appelé son dépôt. Le PVC est un problème fort important en recherche opérationnelle. Il peut être utilisé directement ou comme sous-modèle dans la résolution de nombreux problèmes pratiques concernant l'expédition de véhicules. Pour ce dernier type de problèmes toutefois, le PVC a avantage à être généralisé dans le but d'être adapté à des situations plus complexes. En particulier, le nombre de commis ou véhicules dont on doit établir les routes est dans la réalité habituellement supérieur à 1 et quelquefois même variable. En outre ces routes sont souvent assujetties à plusieurs types de contraintes et peuvent débuter et se terminer à plus d'un dépôt dont la localisation n'est pas nécessairement connue a priori. Les problèmes suivants ont été étudiés de façon à tenir compte des constatations précédentes:

1. le problème des m voyageurs de commerce;
2. le problème des m voyageurs de commerce assujettis à une restriction de capacité;
3. le problème des m voyageurs de commerce avec plusieurs dépôts;
4. le problème des m voyageurs de commerce avec un dépôt variable.

Ce texte contient six chapitres. Le premier contient une description générale de la recherche effectuée. Le deuxième résume la littérature concernant le PVC. Chacun des chapitres 3 jusqu'à 6 concerne un des problèmes précédents. On peut diviser en trois parties principales le matériel qui y est présenté:

1. résumé de la littérature au sujet du problème traité;
2. présentation d'un nouveau modèle pour ce problème;
3. présentation d'un nouvel algorithme optimal pour ce problème.

Les nouveaux modèles décrits sont tous des extensions du modèle de Dantzig, Fulkerson et Johnson pour le PVC et, à notre connaissance, c'est la première fois que ce modèle est généralisé de façon à traiter d'un dépôt variable ou d'une contrainte de capacité sur les commis. En plus de prouver la validité des modèles présentés, les différentes relations qui existent entre ces derniers et d'autres formulations déjà parues dans la littérature sont étudiées.

Les algorithmes construits sont tous basés sur la même approche générale résolvant un problème linéaire tout entier: initialement la plupart des contraintes du modèle sont relaxées et, par la suite, certaines d'entre elles sont réintroduites explicitement dans le modèle. L'outil principal des calculs est la méthode du simplex utilisée conjointement avec une méthode d'hyperplans de coupe ou une méthode de "Branch-and-Bound" de façon à imposer l'intégrité sur les variables. Les travaux décrits ici constituent en fait une validation de ce type d'approche pour résoudre des problèmes concernant l'expédition des véhicules. Pour le problème des m voyageurs de commerce et celui des m voyageurs de commerce assujettis à une restriction de capacité, les algorithmes présentés se sont avérés compétitifs sinon supérieurs à ceux déjà existants. A notre avis, la littérature ne contient aucun algorithme optimal pour les deux autres généralisations du PVC considérées. Ces problèmes étaient habituellement perçus comme trop difficiles à résoudre de façon exacte. A l'aide des algorithmes développés, des problèmes avec m voyageurs de commerce et plusieurs dépôts contenant 60 villes de même que des problèmes avec m voyageurs de commerce et un dépôt variable contenant 50 villes ont été résolus optimalement.

CHAPITRE 1

DESCRIPTION DE NOTRE RECHERCHE

1. DESCRIPTION DE NOTRE RECHERCHE

1.1 Quatre généralisations du problème du voyageur de commerce

Un des problèmes les plus étudiés au cours des deux dernières décennies en recherche opérationnelle est sans contredit le problème du voyageur de commerce(1) [8,13,17] identifié dans ce texte par PVC: il s'agit du problème auquel fait face un voyageur de commerce qui doit, en partant d'une ville appelée le dépôt, visiter (n-1) autres villes, chacune exactement une fois, puis revenir au dépôt de façon à minimiser la distance totale parcourue. Nous emploierons indifféremment dans ce texte les termes "commis", "commis-voyageur", "voyageur de commerce" ou même "véhicule". Le dépôt constituera toujours un point de départ et d'arrivée pour un commis. On dira qu'un commis est "basé" à un dépôt s'il doit y commencer et terminer son périple. Il sera fréquent de considérer une ville comme un client auquel le commis doit rendre un certain service.

Les applications du PVC sont nombreuses [50,83] et tout algorithme pour ce problème peut s'avérer très utile en pratique. Toutefois, malgré la simplicité apparente du PVC, sa résolution de façon exacte a constitué pendant longtemps un défi majeur pour les chercheurs. Ce n'est que récemment qu'apparaissaient dans la littérature des algorithmes optimaux(2) pouvant résoudre des PVC de 100 villes et plus (voir par exemple [2,15,27,54,92,98,114]). Ces succès récents nous ont incités, de même que plusieurs autres chercheurs, à entreprendre la construction d'algorithmes optimaux pour quelques généralisations du PVC à partir d'idées ou techniques d'optimisation efficaces pour ce problème. L'intérêt de tels travaux réside surtout dans le fait que pour un grand nombre de situations pratiques, notamment celles qui concernent le transport physique de marchandises ou de personnes, le modèle du PVC est inadéquat. Souvent les routes(3) de plus d'un commis doivent être construites à

---

(1) En anglais: "The travelling salesman problem".

(2) Il s'agit de méthodes (algorithmes) permettant de trouver un optimum global par opposition à des méthodes (algorithmes) approximatives ou heuristiques.

(3) Une route est définie par un ordre selon lequel le commis doit visiter certaines villes. On parlera de la route ou du tour d'un commis.

partir de plusieurs dépôts et quelquefois même des contraintes supplémentaires, spécifiques au problème considéré, viennent restreindre le nombre de routes possibles.

Le PVC et ses quatre généralisations considérées ici font partie de deux ensembles importants de problèmes en recherche opérationnelle dont les appellations anglaises sont "vehicle routing and scheduling problems" [11,12] ainsi que "location-routing problems" [86] que nous traduirons respectivement par "les problèmes d'expédition de véhicules" (PEV) et "les problèmes de localisation et d'expédition" (PLE). Dans le cas des PEV, les routes d'un ou plusieurs véhicules, partant d'un ou plusieurs dépôts, doivent être établies de façon à respecter certaines contraintes opérationnelles(1) tout en minimisant la somme des coûts de transport. Le nombre de véhicules utilisés à chaque dépôt peut être fixé à l'avance ou non. De même les caractéristiques de ces véhicules peuvent être identiques ou différentes. Dans [11,12] on retrouve de façon plus détaillée les différentes composantes et applications des PEV.

Les problèmes du type PLE peuvent être considérés comme une généralisation des PEV au sens où les points de départ et d'arrivée des véhicules ne sont plus supposés connus a priori et où leurs localisations doivent être effectuées simultanément avec la construction des routes des véhicules. La recherche concernant les PLE en est encore à un stage préliminaire et, à notre connaissance, très peu de méthodes exactes ont été construites pour ce type de problème. Madsen [86] a publié récemment un résumé des connaissances actuelles dans ce domaine.

Trois des quatre généralisations du PVC que nous étudierons constituent des cas particuliers des PEV. Considérons m voyageurs de commerce et supposons pour l'instant qu'ils sont tous basés au même dépôt dont la localisation est connue(2). Les (n-1) autres villes du problème doivent recevoir exactement une visite d'un seul commis. En supposant

---

(1) On donnera au chapitre 4 quelques exemples de ces contraintes.

(2) Parmi les n villes du problème, celle qui sert de dépôt est identifiée.

que la route d'un commis peut contenir un nombre quelconque de villes, le problème de trouver les m routes (une par commis) de coût total minimal s'appelle le problème des m voyageurs de commerce[1] (m-PVC). Le nombre de routes est fixe ou variable et on suppose connu le coût[2] de transport entre chaque paire de villes, coût invariant[3] selon le commis utilisé. Le m-PVC sera analysé au chapitre 3.

Reprenons le contexte du m-PVC mais supposons cette fois que chaque ville i autre que le dépôt demande qu'une certaine quantité d_i d'un bien donné lui soit livrée et dénotons par D la capacité de livraison de chacun des m véhicules. La demande totale des villes sur leur route devra donc être inférieure ou égale à D. Le problème des m voyageurs de commerce assujettis à une restriction de capacité comme la précédente sera abrégé par c-m-PVC[4] et étudié au chapitre 4.

Au chapitre 5, le problème considéré emprunte encore une fois le contexte du m-PVC mais permet l'utilisation de p (>1) dépôts parmi les n villes. On exige qu'au moins un commis commence et termine une route à chacun des p dépôts dont les emplacements sont connus à l'avance. En outre, une route ne pourra contenir plus d'un dépôt. L'abréviation utilisée pour ce problème est (p,m)-PVC. Notons qu'aucune restriction de capacité n'est imposée sur les routes des commis.

La dernière généralisation du PVC traitée ici (chapitre 6) est un cas particulier des problèmes du type PLE. Il s'agit d'un m-PVC avec dépôt variable où la localisation du dépôt parmi les n villes doit être effectuée simultanément avec la construction des m routes de façon à optimiser les coûts de transport et de localisation. Le nombre de commis utilisés est fixe et l'abréviation de ce problème est v-m-PVC.

Le tableau 1.1 résume les principales caractéristiques du PVC et de ses quatre généralisations étudiées ici.

---

(1) En anglais: "The multisalesmen problem".

(2) On préférera le terme "coût" au lieu de distance dans la suite du texte.

(3) Dans certains PEV cette hypothèse est relaxée.

(4) En anglais: "The truck dispatching problem".

TABLEAU 1.1

Description du PVC et de ses 4 généralisations considérées

Problème	Chapitre	Nombre de commis	Restriction de capacité	Nombre de dépôts	Localisation(s) du (des) dépôt(s)
PVC	2	1	absente	1	connue
m-PVC	3	m	absente	1	connue
c·m-PVC	4	m	présente	1	connue
(p,m)-PVC	5	m (1)	absente	p(>1)	connues
v·m-PVC	6	m	absente	1	inconnue

Vu l'étroite relation existant entre nos quatre généralisations et le PVC, un résumé des principaux résultats concernant ce dernier problème sera donné au chapitre 2.

1.2 Principales étapes de notre recherche

Nos travaux sur chacune des généralisations du PVC décrites en 1.1 comportent trois étapes principales:

1. construction d'un modèle mathématique;
2. construction d'une méthode exacte de résolution du modèle établi;
3. compilation de résultats de tests sur ordinateur.

En général, le contenu des chapitres 3, 4, 5 et 6 sera donc le suivant:

1. résumé de la littérature;
2. description des étapes 1, 2 et 3 précédentes;
3. comparaison de notre modèle et de notre algorithme versus ceux déjà existants.

---

(1) Au total pour les p dépôts.

Le chapitre 2 contient peu de matériel nouveau et constitue surtout un résumé de la littérature au sujet du PVC. Notons de plus que notre résumé des méthodes résolvant un problème donné portera principalement sur les algorithmes optimaux pour ce problème car tel est le type de méthodes qui nous intéresse.

Plusieurs points communs relient nos résultats des chapitres 3 jusqu'à 6. Tout d'abord à l'étape 1 tous les modèles mis au point sont nouveaux et constituent des généralisations du modèle de Dantzig et al. [28] pour le PVC, modèle que nous verrons au chapitre 2. Chacun de ces modèles est de la forme d'un problème de programmation linéaire tout entier avec variables principalement binaires, contenant un très grand nombre de contraintes. De plus la méthodologie générale utilisée à l'étape 2 pour les résoudre est la même: la majorité des contraintes sont initialement relaxées et certaines d'entre elles sont par la suite réintroduites explicitement dans le modèle. Une méthode de "Branch-and-Bound" [40,78,95] ou bien d'hyperplans de coupe [53] servira à maintenir l'intégrité. Nos travaux constituent donc une validation de ce type d'approche à certains problèmes du type PEV ou PLE et font suite aux travaux [27,28,54,91,92,98] concernant le PVC. Les résultats des chapitres 3, 5 et 6 ont été publiés respectivement dans [74,76,75] et le chapitre 4 a fait l'objet d'une présentation [73].

1.2.1 Génération des problèmes-tests aléatoires

Finalement à l'étape 3, tous les tests sur ordinateur ont été effectués à l'aide de l'ordinateur CYBER 173 de l'Université de Montréal avec un compilateur FTN. Tous les problèmes-tests, sauf un[1], ont été engendrés aléatoirement à l'aide d'une distribution uniforme sur (0,100). La modalité de la génération de la matrice des coûts de transport $C = (c_{ij})$ dépend du type considéré: symétrique[2] ou asymétrique, euclidien[3] ou non-euclidien. Dans les cas asymétrique et symétrique non-euclidien, les quantités c_{ij} sont engendrées directement. Dans le cas symétrique, des points dans

---

(1) Le problème de Dantzig et Ramser [29] pour le c.m-PVC.

(2) $c_{ij} = c_{ji}$ pour tout i,j.

(3) $c_{ij} \leq c_{ik}+c_{kj}$ pour tout i,j,k.

$(0,100)^2$ sont tout d'abord engendrés. Le coût c_{ij} est ensuite posé égal à la distance euclidienne entre les points i et j.

1.3 Notation et terminologie générales

En plus de la notation usuelle de la théorie des ensembles, nous utiliserons:

$\lceil x \rceil$	le plus petit entier supérieur ou égal à x;
$\lfloor x \rfloor$	le plus grand entier inférieur ou égal à x;
C_{pxq}	une matrice de p rangées et q colonnes;
$P, P^1, P^2, \ldots$	des problèmes de programmation mathématique;
F(P)	le domaine réalisable (admissible) de P;
V(P)	la valeur que prend l'objectif de P pour une de ses solutions optimales. V(P) sera appelée la valeur de P. Si $F(P) = \phi$, alors:

$$V(P) = \begin{cases} -\infty \text{ pour une maximisation} \\ +\infty \text{ pour une minimisation} \end{cases}$$

1.4 Terminologie de la théorie des graphes[1] [10]

1.4.1 Terminologie associée aux graphes orientés

Graphe orienté

Un graphe orienté G est constitué par un couple orienté (N,A) où:

1. N est un ensemble d'éléments appelés les sommets ou noeuds de G;
2. A est un ensemble de couples ordonnés (i,j) appelés les arcs de G où i et j sont des éléments distincts de N. On peut écrire $A \subset N \times N$.

On dénote un tel graphe par G = (N,A).

---

(1) Dans certains textes sur la théorie des graphes, notamment [10], la définition d'un graphe est plus générale que la nôtre. Nous avons voulu particulariser notre définition aux problèmes traités sans toutefois altérer l'essence de ces définitions.

Dans la section 1.4.1 le terme graphe veut dire graphe orienté. La terminologie suivante est fréquemment rencontrée en conjonction avec un graphe:

Extrémité initiale d'un arc

Le sommet i de l'arc (i,j) constitue son extrémité initiale. On dira aussi sommet initial.

Extrémité terminale d'un arc

Le sommet j de l'arc (i,j) constitue son extrémité terminale. On dira aussi sommet terminal.

Ordre d'un graphe

Le nombre de sommets d'un graphe est appelé son ordre. On dira que le graphe $G = (N,A)$ est d'ordre n si $|N| = n$.

Graphe complet

Un graphe $G = (N,A)$ d'ordre n est complet si $|A| = n(n-1)$, c'est-à-dire si tous les arcs possibles sont inclus dans A. On peut alors écrire $A = \{(i,j) : i \in N, j \in N, i \neq j\}$.

Successeur d'un sommet

Un sommet $j \in N$ est un successeur du sommet $i \in N$ dans le graphe $G = (N,A)$ si $(i,j) \in A$.

Prédécesseur d'un sommet

Un sommet $i \in N$ est un prédécesseur du sommet $j \in N$ dans le graphe $G = (N,A)$ si $(i,j) \in A$.

Degré intrant[1] d'un sommet

Le degré intrant d'un sommet dans un graphe est le nombre de prédécesseurs qu'il possède dans ce graphe.

---

(1) Dans un graphe complet d'ordre n le degré intrant et le degré extrant de chacun de ses sommets est égal à (n-1).

Degré extrant d'un sommet

Le degré extrant d'un sommet dans un graphe est le nombre de successeurs qu'il possède dans ce graphe.

Arc incident à un sommet

L'arc (i,j) est dit incident au sommet i vers l'extérieur et incident au sommet j vers l'intérieur.

Arcs adjacents

Deux arcs sont dits adjacents s'ils ont au moins une extrémité commune, soit l'extrémité initiale, soit l'extrémité terminale.

Chaîne d'un graphe

Une chaîne dans le graphe G = (N,A) est une séquence de $q(>0)$ arcs de A de la forme $((i_1,j_1), (i_2,j_2),\ldots,(i_q,j_q))$ telle que chaque arc (i_p,j_p) avec $2 \leq p \leq q-1$ possède une extrémité en commun avec l'arc précédent dans la séquence et l'autre extrémité en commun avec l'arc suivant dans la séquence. On dira qu'une telle chaîne relie les sommets i_1 et i_q qui en sont respectivement le sommet initial et terminal.

Chemin d'un graphe

Un chemin est une chaîne $((i_1,j_1), (i_2,j_2),\ldots,(i_q,j_q))$ telle que $j_p = i_{p+1}$ pour $p = 1,\ldots,q-1$. Pour faciliter l'écriture nous dénoterons un chemin par le vecteur des sommets qui en font partie. Ainsi le chemin $\mu = (i_1,\ldots,i_q)$ correspond à la suite d'arcs $((i_1,i_2),(i_2,i_3),\ldots,(i_{q-1},i_q))$. Les sommets i_1 et i_q sont respectivement les sommets initial et terminal du chemin μ. On dira que c'est un chemin de i_1 vers i_q.

Chemin simple d'un graphe

Un chemin $\mu = (i_1,\ldots,i_q)$ est dit simple si les sommets $i_1,\ldots,i_{q-1}$ sont distincts. A partir de maintenant le terme "chemin" est employé pour chemin simple.

Circuit d'un graphe

Un chemin $\mu = (i_1,\ldots,i_q)$ est un circuit si $i_1 = i_q$.

Circuit hamiltonien d'un graphe

Un circuit hamiltonien dans un graphe $G = (N,A)$ d'ordre n est un circuit contenant n arcs éléments de A. Un circuit hamiltonien sera appelé aussi un tour dans le chapitre 2.

Sous-tour d'un graphe

Un sous-tour dans un graphe $G = (N,A)$ d'ordre n est un circuit contenant p arcs de A avec $2 \leq p \leq n-1$.

Sous-graphe d'un graphe

Un sous-graphe $G' = (N',A')$ du graphe $G = (N,A)$ est un graphe tel que $N' \subseteq N$ et $A' \subseteq A$.

Graphe connexe

Un graphe $G = (N,A)$ est connexe si pour toute paire de sommets distincts i et j dans N il existe une chaîne de G avec i comme son sommet initial et j comme son sommet terminal.

Composante connexe d'un graphe

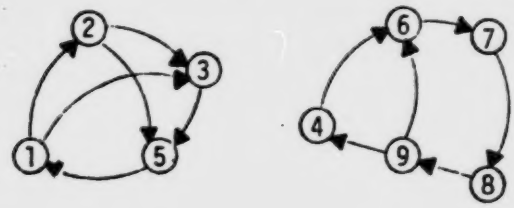
Une composante connexe d'un graphe $G = (N,A)$ est un ensemble de sommets dans N avec la particularité suivante: deux sommets de G font partie de la même composante connexe si et seulement si il existe une chaîne de G les reliant. Un graphe connexe ne possède donc qu'une seule composante connexe tandis qu'un graphe non connexe peut être divisé en au moins deux composantes connexes disjointes (les ensembles de sommets associés ont une intersection vide). Un graphe non connexe ne peut contenir de circuit hamiltonien.

La figure 1.1 suivante illustre un graphe orienté en utilisant la convention que le sommet $i \in N$ du graphe $G = (N,A)$ est symbolisé par

(i) et l'arc (i,j) ∈ A est représenté par la flèche (i)→(j).

FIGURE 1.1

Exemple d'un graphe orienté G = (N,A)



Les constatations suivantes peuvent être faites au sujet du graphe de la figure 1.1:

- l'ordre du graphe est 9;
- ce graphe n'est pas complet ni connexe;
- le degré intrant du sommet 3 est 2 et son degré extrant est 1;
- les arcs (2,3) et (1,3) sont adjacents;
- la séquence d'arcs ((1,3), (2,3), (2,5)) est une chaîne de ce graphe;
- la séquence d'arcs ((1,2), (2,3), (3,5)) est un chemin simple de ce graphe, chemin dénoté (1,2,3,5);
- le chemin (4,6,7,8,9,4) est un circuit de ce graphe;
- l'ensemble de sommets {1,2,3,5} constitue une composante connexe de ce graphe;
- le circuit (4,6,7,8,9,4) constitue un circuit hamiltonien sur le sous-graphe de G noté G' = (N',A') où N' = {4,6,7,8,9} et où A' = {(4,6), (6,7), (7,8), (8,9), (9,6), (9,4)}.

1.4.2 Terminologie associée aux graphes non-orientés

Graphe non-orienté

Un graphe non-orienté G = (N,A) est formé par:

1. un ensemble N d'éléments appelés ses sommets (ou noeuds);
2. un ensemble A de couples non-ordonnés (i,j) appelés ses arêtes où i et j sont des sommets distincts de N.

Dans un graphe non-orienté les couples (i,j) et (j,i) représentent donc la même entité, notamment l'arête (i,j)[1] tandis que dans un graphe orienté ces couples correspondent à deux arcs différents, le premier orienté de i vers j et le deuxième orienté dans le sens inverse. On dira que l'arête (i,j) peut être traversée selon les deux orientations possibles. Plusieurs notions associées à un graphe orienté n'ont donc plus leur place dans le contexte d'un graphe non-orienté, notamment les notions "extrémité initiale", "extrémité terminale", "successeur", "prédécesseur", "degré intrant", "degré extrant", etc. La notion d'ordre est identique que le graphe soit orienté ou non. Les définitions suivantes nous seront utiles en conjonction avec un graphe non-orienté. Plusieurs termes sont communs aux deux types de graphe même si leur définition diffère d'un type à l'autre. Dans cette section "graphe" signifie graphe non-orienté.

Extrémité d'une arête

Les sommets i et j sont les deux extrémités de l'arête (i,j).

Arête incidente à un sommet

L'arête (i,j) est incidente au sommet i et au sommet j.

Arêtes adjacentes

Deux arêtes sont dites adjacentes si elles possèdent une extrémité commune.

---

(1) L'arête dont les extrémités sont les sommets i et j sera dénotée par (i,j) si $i < j$ et par (j,i) sinon.

Degré d'un sommet

Le degré d'un sommet dans un graphe $G = (N,A)$ est le nombre d'arêtes de A qui lui sont incidentes.

Graphe complet

Un graphe $G = (N,A)$ d'ordre n est dit complet si $|A| = \frac{1}{2} n(n-1)$, c'est-à-dire si toutes les arêtes possibles sont incluses dans A. On peut alors écrire $A = \{(i,j) : i \in N, j \in N, i < j\}$.

Sous-graphe d'un graphe

Un sous-graphe d'un graphe $G = (N,A)$ est un graphe $G' = (N',A')$ où $N' \subseteq N$ et $A' \subseteq A$.

Chaîne d'un graphe

Une chaîne d'un graphe $G = (N,A)$ est un ensemble $U \subseteq A$ dont les éléments peuvent être ordonnés sous la forme d'une séquence[1] $((i_1,j_1), (i_2,j_2),\ldots,(i_q,j_q))$ possédant les propriétés suivantes:

1. $q > 0$
2. pour $q > 1$ et $p = 2,\ldots,q-1$:
   (i_p,j_p) possède une extrémité en commun avec (i_{p-1},j_{p-1}) et l'autre extrémité en commun avec (i_{p+1}, j_{p+1}).

Soit i l'extrémité de (i_1,j_1) qui n'est pas une extrémité de (i_2,j_2). Soit j l'extrémité de (i_q,j_q) qui n'est pas une extrémité de (i_{q-1},j_{q-1}). On dira qu'une telle chaîne relie les sommets i et j.

Chaîne simple d'un graphe et cycle d'un graphe

Soit $G' = (N',A')$ le sous-graphe de $G = (N,A)$ associé à une chaîne U de G. Le graphe G' est défini de la façon suivante:

$N' = \{i \in N : i$ est une extrémité d'au moins une arête de $U\}$

$A' = U$

---

(1) Deux séquences différentes pour un même ensemble U sont considérées comme des chaînes distinctes.

La chaîne U est dite simple si G' satisfait aux deux propriétés suivantes:

1. exactement deux sommets de G' ont un degré de 1 dans G';
2. tous les autres sommets de G' ont un degré de 2 dans G'.

La chaîne U est appelée un cycle de G si tous les sommets de G' ont un degré de 2 dans G'.

Cycle hamiltonien[1] d'un graphe

Un cycle hamiltonien d'un graphe G = (N,A) d'ordre n est un cycle de G contenant exactement n arêtes.

Sous-tour

Un sous-tour d'un graphe G = (N,A) d'ordre n est un cycle de G contenant p arêtes avec $3 \leq p \leq n-1$.

Graphe connexe

Un graphe G = (N,A) est dit connexe si pour toute paire de sommets distincts i et j dans N il existe une chaîne de G reliant i et j.

Composante connexe d'un graphe

Une composante connexe d'un graphe G = (N,A) est un ensemble de sommets de G avec la particularité suivante: deux sommets de G font partie de la même composante connexe si et seulement si il existe une chaîne de G les reliant. Tout comme dans le cas orienté, un graphe connexe ne contient qu'une seule composante connexe. De plus un graphe non connexe ne peut posséder un cycle hamiltonien.

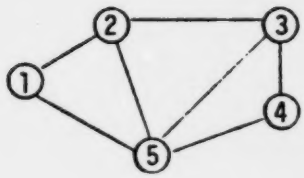
La figure 1.2 suivante illustre un graphe non-orienté en représentant le sommet i par "(i)" et l'arête (i,j) par "(i)—(j)"

FIGURE 1.2 / PAGE SUIVANTE

(1) Un tel cycle sera aussi appelé un tour dans le chapitre 2.

FIGURE 1.2

Exemple d'un graphe non-orienté



Au sujet du graphe de la figure 1.2 on fera remarquer que:

- son ordre est 5;
- il est connexe mais non complet;
- le degré du sommet 2 est 3 tandis que celui du sommet 1 est 2;
- les arêtes (2,3) et (3,5) sont adjacentes;
- (3,5) est incidente aux sommets 3 et 5;
- {(1,2), (2,3), (3,5), (2,5)} est une chaîne (non simple) reliant les sommets 1 et 2;
- {(1,2), (2,5), (1,5)} est un cycle non hamiltonien.

1.4.3 Correspondance entre un graphe et un problème de type PLE ou PEV

A tout problème de type PLE ou PEV on peut associer:

1. un ensemble $N = \{1,\ldots,n\}$ de villes;
2. une matrice $C_{nxn} = (c_{ij})$[1] des coûts de transport entre chaque couple ordonné de villes i et j.

Lorsque C est symétrique, le problème est dit symétrique sinon il est asymétrique. L'association entre un problème donné P et un graphe $G = (N,A)$ se fait à l'aide des correspondances suivantes:

1. P asymétrique ⟷ G orienté
2. P symétrique ⟷ G non-orienté
3. ville $i \in N$ ⟷ sommet $i \in N$

(1) $c_{ii} = \infty$ pour tout i.

4. Trajet[1] de i vers j ⟷ arc ou arête (i,j)
5. Trajet de j vers i ⟷ arc (j,i) ou arête (i,j)

On appellera c_{ij} le poids ou coût de l'arc ou arête (i,j). Le poids d'un graphe G est défini comme la somme des poids de ses arcs ou arêtes. Tout problème P de type PEV ou PLE auquel est associé un graphe G peut se définir comme la recherche d'un sous-graphe de G de poids minimal possédant une caractéristique donnée. Ainsi le PVC symétrique se décrit comme la recherche dans un graphe G d'un sous-graphe de poids minimal constituant un cycle hamiltonien dans G.

Lorsque le passage direct d'une ville i vers une autre ville j n'est pas permis, c_{ij} prend une valeur arbitrairement grande, l'arc ou arête correspondant à ce trajet n'existe pas dans le graphe G associé au problème et ce dernier n'est pas complet[2]. Tous les problèmes que nous avons testés sur ordinateur sont associés à un graphe G complet.

1.5 Principales variables et notations utilisées dans les modèles mathématiques

Trois types principaux de variables seront utilisés dans les modèles mathématiques présentés dans les prochains chapitres. Il s'agit de:

1. x_{ij} : la variable associée à l'arc ou arête (i,j) indiquant le nombre de fois[3] que le trajet entre les villes i et j est effectué. Pour un problème asymétrique le passage correspondant à x_{ij} est orienté de i vers j. Le coût de la variable x_{ij} est le poids de l'arc ou arête qui lui est associé, notamment c_{ij}. Dans l'écriture x_{ij} on aura toujours $i \neq j$.

---

(1) Il s'agit du passage direct de la ville i à la ville j sans qu'aucune autre ville intermédiaire ne soit visitée.

(2) Dans un tel cas la matrice C ne peut être euclidienne.

(3) Habituellement 0, 1 ou 2.

2. m_k : la variable entière non-négative indiquant le nombre de commis qui suivront une route débutant et se terminant au dépôt k. Lorsque le problème contient un seul dépôt, on utilisera m au lieu de m_1.

3. y_k : la variable binaire indiquant si la ville k est utilisée comme dépôt ($y_k = 1$) ou non ($y_k = 0$).

Les deux premiers types de variables précédents sont fréquemment employés dans les modèles mathématiques pour les PEV et nous seront utiles dans le cas du m-PVC, du c·m-PVC et du (p,m)-PVC. En plus des variables x_{ij} et m_k, des variables du type y_k sont nécessaires pour formuler les PLE et nous nous en servirons en conjonction avec le v·m-PVC.

Pour simplifier l'écriture on dira, avec un certain abus de langage, que la "variable x_{ij}" possède une caractéristique donnée de la théorie des graphes au lieu de dire que "l'arc (ou arête) associé à x_{ij} "possède cette caractéristique. Ainsi, par exemple, la variable x_{ij} est incidente au sommet i selon le sens mentionné précédemment.

Pour un problème asymétrique auquel correspond un graphe orienté complet d'ordre n on aura besoin de n·(n-1) variables du type x_{ij}, c'est-à-dire une pour chaque arc de G. Pour un problème symétrique associé à un graphe non-orienté complet d'ordre n ce nombre est $\frac{1}{2}n(n-1)$ A la rigueur on peut toujours modéliser un problème symétrique comme un problème asymétrique: à chaque arête (i,j) on fait correspondre l'arc (i,j) et l'arc (j,i). Dans ce modèle, les variables x_{ij} et x_{ji}, de coût identique c_{ij}, remplacent la variable unique x_{ij} de coût c_{ij}. Ceci implique toutefois l'utilisation de deux fois plus de variables que nécessaire et s'avère souvent néfaste d'un point de vue algorithmique[1]. Dans ce qui suit les termes "modèle (formulation) symétrique" signifient un modèle où, pour les variables x_{ij}, seules celles avec $i < j$ sont utilisées. De la même façon "modèle asymétrique" veut dire un modèle où il

---

(1) Ceci est particulièrement vrai pour le PVC (voir chapitre 2).

y a autant de variables x_{ij} que de couples ordonnés (i,j) avec $i \neq j$. Il sera toujours sous-entendu qu'un modèle asymétrique peut aussi servir dans le cas symétrique.

Plusieurs constantes seront utilisées conjointement avec les variables précédentes. En voici la liste et leur signification:

p	le nombre de villes constituant des dépôts dans N;
K	l'ensemble des dépôts ($K \subseteq N$, $K \neq \phi$);
f_k	le coût fixe d'utilisation[1] de la ville k comme dépôt;
$\underline{m}$, $\bar{m}$	respectivement, une borne inférieure et supérieure sur la variable m;
$\underline{m}_k$, $\bar{m}_k$	respectivement, une borne inférieure et supérieure sur la variable m_k;
g	le coût fixe d'utilisation d'un commis[2].

1.6 Outils informatiques

1.6.1 Les programmes MIF et BB

Pour implanter nos algorithmes sur ordinateur nous avons modifié les deux programmes suivants:

1. Programme MIF:

   Ce programme est une version en arithmétique entière[3] (construite par Miliotis [90]) du programme MIF contenu dans le code de Land et Powell [71]. Il résout un problème de pro-

---

(1) Coût d'ouverture d'un dépôt à la ville k, coût d'opération d'un dépôt à la ville k, etc.

(2) Dans le cas où le nombre de commis est variable, on peut vouloir tenir compte d'un tel coût en plus des coûts de transport.

(3) Le tableau du simplex est multiplié par le déterminant de la matrice inverse de la matrice de base de façon à ce que tous les calculs du simplex se fassent en arithmétique entière.

grammation linéaire tout entier par l'ajout de coupes de Gomory [53,71].

2. Programme BB:

Ce programme est celui du code de Land et Powell résolvant un problème de programmation linéaire mixte[1] par une méthode de "Branch-and-Bound".

Ces deux programmes ainsi que la sous-routine LP, utilisée conjointement par MIF et BB, seront décrits brièvement dans cette section. Pour plus d'informations nous référons le lecteur à [71].

La sous-routine LP:

Cette sous-routine résout un problème de programmation linéaire par une version modifiée de la méthode du simplex. Elle est conçue de façon à débuter ses itérations à partir d'une solution initiale réalisable ou non et aucune variable artificielle n'est ajoutée au problème. LP peut être ainsi utilisée conjointement avec MIF ou BB qui lui demandent d'optimiser des problèmes à partir d'une solution de base irréalisable.

La structure de données utilisée par LP pour décrire la matrice A des coefficients des contraintes est telle qu'il est facile de la modifier pour tenir compte d'une contrainte supplémentaire. L'ajout de nouvelles variables est toutefois plus difficile à effectuer.

La matrice inverse de la matrice de base est gardée en mémoire de façon compacte. Soit B une matrice m x m associée à une solution de base donnée. En général B contient $(m-k)$[2] vecteurs unitaires correspondant aux (m-k) variables d'écart dans la base. En ordonnant de nou-

---

(1) Un problème de programmation linéaire avec variables entières et variables continues.

(2) $0 \leq m-k \leq m$.

veau les rangées et les colonnes de B on peut écrire B sous la forme:

$$B = \begin{bmatrix} R & 0 \\ S & I \end{bmatrix} \qquad (1.1)$$

où I est une matrice unité d'ordre (m-k).

L'expression de B^{-1} sera alors:

$$B^{-1} = \begin{bmatrix} R^{-1} & 0 \\ -SR^{-1} & I \end{bmatrix} \qquad (1.2)$$

D'après (1.2), la multiplication d'un tableau du simplex par B^{-1} nécessite uniquement la connaissance de R^{-1}, la matrice S étant incluse dans la structure de données correspondante à A. On appellera R^{-1} l'inverse réduit de la base et les contraintes associées à R seront dénommées les contraintes effectives. Celles-ci sont habituellement des contraintes d'inégalité satisfaites à égalité par la solution de base en main [71]. Les contraintes ne correspondant pas à une rangée de R seront dites "ineffectives" et leur variable d'écart est habituellement non nulle dans la solution de base considérée.

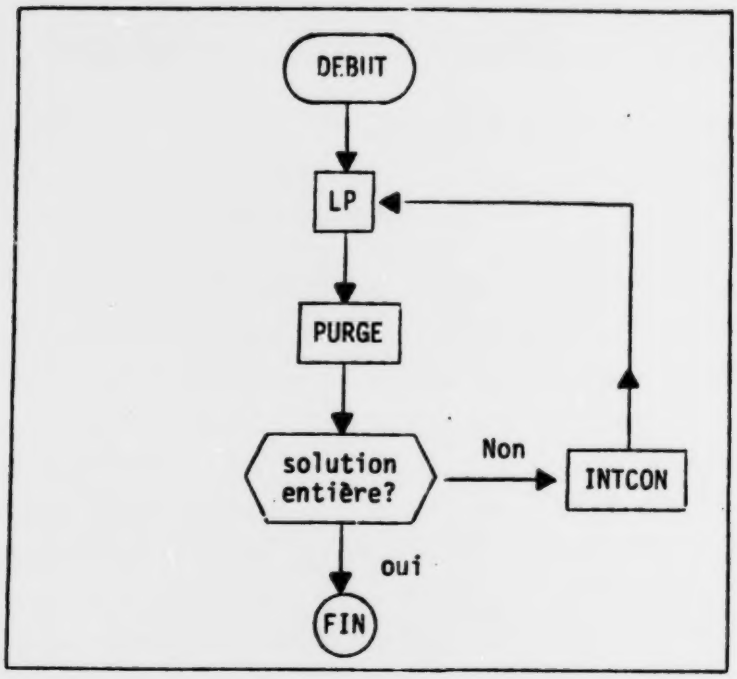
Le programme MIF

Le programme MIF résout un problème linéaire tout entier. La version de Miliotis [90] procède en arithmétique entière mais est pour le reste similaire à celle du code de Land et Powell. En plus de LP, MIF comprend deux autres sous-routines principales, notamment PURGE et INTCON. La figure 1.3 illustre le fonctionnement de MIF:

FIGURE 1.3 / PAGE SUIVANTE

FIGURE 1.3

Programme MIF



Soit P un problème linéaire tout entier et P^1 sa relaxation continue(1). P^1 est résolu par LP. Si x^1, la solution de P^1, est entière on termine sinon INTCON ajoute une ou plusieurs coupes de Gomory à P^1. Le problème obtenu, P^2, est une relaxation de P tel que x^1 est irréalisable pour P^2. LP est de nouveau appelée pour résoudre P^2 en commençant ses itérations avec x^1. On a vu que LP peut être initialisée avec une solution irréalisable et que l'ajout de coupes peut s'effectuer aisément.

Le rôle de la sous-routine PURGE consiste à éliminer toute coupe devenue ineffective après l'exécution de LP. La proportion de coefficients nuls dans une coupe est habituellement faible, à tout le moins pour le type de problèmes que nous avons traités, et PURGE sert essentiellement à économiser l'espace-mémoire nécessaire à l'exécution de MIF. Toutefois il est possible qu'une coupe de pente similaire à

---

(1) La restriction d'intégrité sur les variables est relaxée.

une coupe précédemment éliminée par PURGE doive être engendrée et on y perdrait alors en temps d'exécution.

Notons que le test sur l'intégrité de la solution a été incorporé à INTCON.

Le programme BB

Le programme BB résout par une méthode de "Branch-and-Bound" un problème linéaire mixte. Tous les problèmes étudiés ici sont des problèmes linéaires tout entiers. Il s'avérera toutefois nécessaire dans certains cas d'utiliser BB au lieu de MIF.

L'arbre(1) des sous-problèmes(2) est construit selon une technique LIFO(3) proposée par Land et Doig [70]. La variable de branchement(4) est choisie selon un calcul de pénalités similaire à celui proposé par Little et al. [85]. Seul le critère de sélection de cette variable sera décrit ici. A toute variable x_i prenant la valeur fractionnaire $\bar{x}_i$ dans la solution optimale du sous-problème P présentement considéré on peut associer des pénalités UP_i et $DOWN_i$ telles que:

1. UP_i est une borne inférieure sur la diminution(5) dans V(P) si on ajoute à P la contrainte $x_i = \lceil \bar{x}_i \rceil$;

---

(1) En termes de la théorie des graphes, un arbre est un graphe non-orienté connexe et sans cycle. L'état des calculs dans une méthode de "Branch-and-Bound" peut être représenté par un arbre.

(2) P' est un sous-problème de P si $F(P') \subset F(P)$ et si P et P' ont le même objectif. Habituellement, P' est obtenu à partir de P en ajoutant une ou plusieurs contraintes du type "$x_i = q$". La variable x_i sera dite une variable fixée à la valeur q.

(3) "LAST IN FIRST OUT".

(4) Celle qui sera fixée à une certaine valeur pour créer de nouveaux sous-problèmes.

(5) Il s'agit d'une maximisation dans le code de Land et Powell.

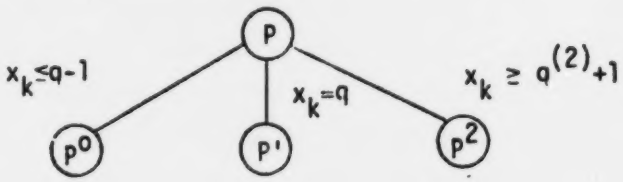
2. $DOWN_i$ est une borne inférieure sur la diminution dans V(P) si on ajoute à P la contrainte $x_i = \lfloor \bar{x}_i \rfloor$.

Soit H l'ensemble des indices des variables restreintes à être entières et prenant une valeur fractionnaire dans la solution optimale de P. L'indice k de la variable de branchement est tel que:

$$\max (UP_k, DOWN_k) = \max_{i \in H} \{\max (UP_i, DOWN_i)\}$$

Si $DOWN_k \geq UP_k$, le nouveau sous-problème considéré[1], noté P^1, est construit à partir de P en adjoignant à ce dernier la contrainte "$x_k = \lceil \bar{x}_k \rceil$". Si $UP_k \geq DOWN_k$, on ajoute à P la contrainte "$x_k = \lfloor \bar{x}_k \rfloor$".

A partir du noeud correspondant à P dans l'arbre des sous-problèmes on crée trois nouveaux problèmes:



P' sera traité en premier et résolu par LP. Les problèmes P^0 et P^2, construits à partir de P en lui ajoutant respectivement les contraintes "$x_k \leq q-1$" et "$x_k \geq q+1$", sont mis dans la liste des sous-problèmes candidats. De plus une borne supérieure sur $V(P^0)$ et $V(P^2)$ est calculée. Le critère de sélection de la variable de branchement est tel que ces bornes seront probablement faibles et P^0 et/ou P^2 pourra être rapidement éliminé des considérations. On dira que P^0, P^1 et P^2 sont à un niveau supérieur[3] à celui de P. Les noeuds associés à P^0 et à P^2 sont appelés des noeuds pendants[4].

---

(1) Ce problème sera résolu par LP.

(2) $q = \begin{cases} \lfloor \bar{x}_k \rfloor \text{ si } UP_k \geq DOWN_k \\ \lceil \bar{x}_k \rceil \text{ sinon} \end{cases}$

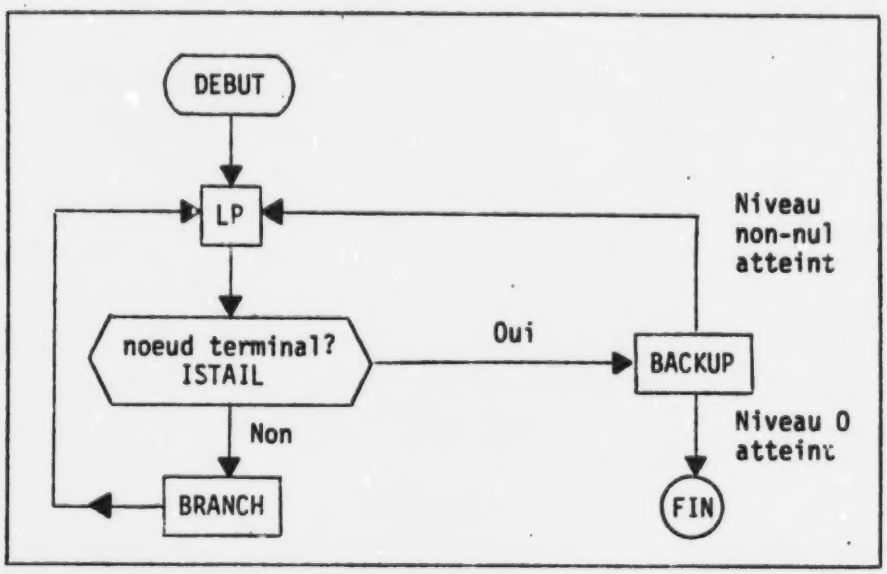
(3) Le premier niveau de l'arbre est le niveau 0.

(4) "Dangling nodes".

Les principales sous-routines de BB sont LP (déjà décrite), ISTAIL, BRANCH et BACKUP. Dans la figure 1.4 on indique l'ordre d'appel de ces sous-routines entre elles.

FIGURE 1.4

Programme BB



Une fois que le sous-problème présentement considéré, noté P, a été résolu par LP, la sous-routine ISTAIL vérifie s'il est nécessaire de créer de nouveaux sous-problèmes à partir de P ou bien si un noeud terminal a été atteint. Dans ce cas, la sous-routine BRANCH est appelée et dans le cas contraire BACKUP est exécutée. Cette dernière choisit essentiellement un nouveau sous-problème à traiter parmi les noeuds pendants tandis que BRANCH effectue les calculs associés à la sélection de la variable de branchement et à la création de 3 nouveaux sous-problèmes (i.e. P^0, P^1 et P^2) dont le prochain qui sera résolu par LP (P^1).

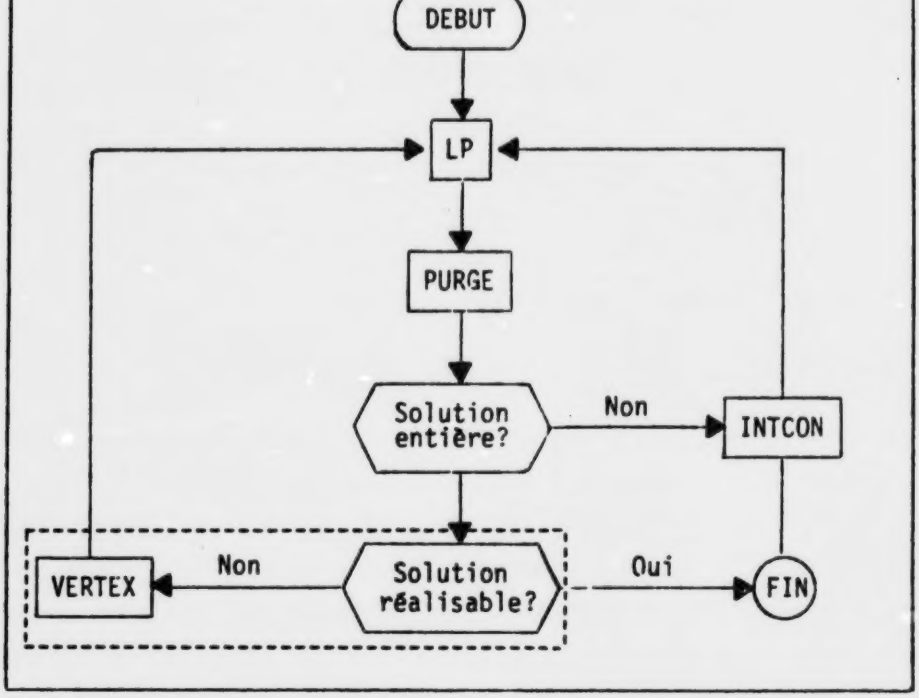
1.6.2 Principales modifications apportées à MIF et BB

Les programmes MIF et BB ont été modifiés dans le but d'être

utilisés dans le cadre de la procédure suivante. Soit P un problème linéaire tout entier, P^1 un problème obtenu en relaxant certaines contraintes de P et P^2 la relaxation continue de P^1. Le problème P peut être résolu en modifiant MIF de la façon décrite à la figure 1.5.

FIGURE 1.5

Modification type effectuée dans MIF



La sous-routine LP résout P^2 initialement et INTCON ajoute des coupes à ce problème tant que la solution obtenue n'est pas entière. Une fois l'intégrité sur les variables obtenue, la solution en main n'est pas nécessairement optimale car elle peut ne pas satisfaire les contraintes de P qui ont été relaxées. Il faut donc insérer dans MIF

une sous-routine, nommée VERTEX[1], qui:

1. vérifie la réalisabilité de la solution en main[2];
2. adjoint à P une ou plusieurs contraintes non satisfaites si nécessaire.

Si VERTEX n'engendre pas de contraintes supplémentaires, l'optimalité est alors atteinte sinon une nouvelle relaxation du type P^2 est obtenue pour P et les calculs sont repris avec cette dernière. Nous avons utilisé une procédure similaire pour résoudre le m-PVC et le (p,m)-PVC. Notons que l'ordre d'appel des sous-routines INTCON et VERTEX est parfois interchangeable comme on l'explicitera plus loin. Dantzig et al. [28] et Martin [89] sont à notre connaissance les premiers à avoir entrepris la résolution du PVC par un algorithme semblable. Ils devaient toutefois effectuer manuellement l'ajout de contraintes au modèle puis réoptimiser dans des passages différents sur ordinateur. Miliotis [92] a évité cette difficulté à l'aide du programme MIF de Land et Powell pour construire une méthode exacte de résolution des PVC symétriques qui s'avère encore maintenant une des plus efficaces. Cette méthode sera décrite au chapitre 2.

Dans le cas où il est préférable de maintenir l'intégrité par une méthode "Branch-and-Bound", les modifications à effectuer dans le programme BB sont similaires à celles apportées à MIF et seront décrites aux chapitres 4 et 6 concernant respectivement le c.m-PVC et le v.m-PVC.

Il est intéressant de constater combien la structure de MIF et BB est particulièrement bien adaptée aux algorithmes que nous avons construits:

1. souplesse de LP quant au type de solution de base initiale;
2. ajout facile de contraintes supplémentaires dans la structure des données;

---

(1) La sous-routine qui vérifiera la réalisabilité des solutions obtenues par rapport aux contraintes relaxées s'appellera VERTEX dans tous nos algorithmes.

(2) Dans la figure 1.5, on a séparé ce test de la sous-routine VERTEX.

3. structure modulaire permettant l'adjonction de nouvelles sous-routines sans modifications majeures;

4. utilisation d'une matrice inverse réduite dont la taille peut demeurer relativement petite par rapport au nombre total de contraintes[1].

Il nous faut souligner cependant que MIF et BB ont été construits dans le but de résoudre des problèmes linéaires tout entiers ou mixtes généraux. Or tous les modèles traités dans ce texte contiennent en majorité des variables binaires. La programmation de MIF et BB n'a pas été modifiée pour tenir compte de cette particularité même si l'efficacité des algorithmes élaborés aurait pu ainsi être quelque peu améliorée.

A part l'ajout de VERTEX[2] dans MIF et BB, plusieurs autres modifications ont été apportées à ces deux programmes selon l'extension du PVC considéré. Elles sont décrites aux endroits appropriés dans les prochains chapitres.

---

(1) Lorsque peu d'entre elles sont effectives simultanément dans les solutions de base rencontrées.

(2) La forme de cette sous-routine varie selon la généralisation considérée.

CHAPITRE 2

LE PROBLEME DU VOYAGEUR DE COMMERCE (PVC)

2. LE PROBLEME DU VOYAGEUR DE COMMERCE

2.1 Introduction

Le PVC possède de nombreuses ramifications théoriques dans plusieurs domaines de la recherche opérationnelle. Ainsi, en termes de la théorie des graphes, il constitue la recherche dans un graphe orienté (non-orienté) du circuit (cycle) hamiltonien de coût minimal. Dans la théorie des matroïdes [34] le PVC est défini par le problème de trouver l'intersection de poids minimal entre 3 matroïdes [79]. Finalement, de façon plus combinatoire, le PVC peut se décrire par le calcul d'une permutation cyclique de coût minimal: soit $p = (i_1,\ldots,i_n)$ une permutation des entiers $(1,2,\ldots,n)$ et P l'ensemble des permutations possibles; résoudre le PVC est équivalent à résoudre le problème suivant:

$$\underset{p\in P}{\text{Minimiser}} \left\{ \left[\sum_{j=1}^{n-1} c_{i_j,i_{j+1}} \right] + c_{i_n,i_1} \right\} \qquad (2.1)$$

Les définitions précédentes du PVC suggèrent quelques approches différentes pour aborder sa résolution. Aucune d'entre elles toutefois ne conduit à un algorithme optimal dont le nombre d'opérations est borné supérieurement par un polynôme[1] (en termes du nombre de villes, d'arêtes ou d'arcs, etc.). En fait il est peu probable qu'un tel algorithme existe car le PVC fait partie de la classe des problèmes NP-complets comme l'a démontré Karp [67] en 1970. Même le problème de trouver une ε-approximation[2] du PVC est un problème NP-complet [107]. Ainsi, malgré sa simplicité apparente, le PVC s'est avéré un problème très difficile à résoudre optimalement. En 1962, Bellman [5] ainsi que Held et Karp [59] proposèrent des méthodes exactes pour le PVC basées sur la programmation dynamique. Ces algorithmes ont résolu des problèmes de 13 villes en un temps d'exécution

---

(1) Un algorithme optimal possédant cette propriété pour un problème donné est appelé un algorithme polynomial pour ce problème.

(2) Un tour dont le coût est à "ε" près du coût du tour optimal.

élevé et avec l'utilisation de beaucoup d'espace-mémoire. La solution optimale de quelques PVC de plus grande taille avait été trouvée précédemment, mentionnons entre autres [28], mais ceci constituait l'exception plutôt que la règle.

Initialement donc les chercheurs ont surtout concentré leurs efforts à la construction d'heuristiques pour le PVC. Cependant, la mise au point de nouveaux modèles pour ce problème conjointement avec la découverte de nouvelles techniques d'optimisation(1) de problèmes linéaires tout entiers permirent d'améliorer grandement la performance des algorithmes optimaux pour le PVC. Aujourd'hui des PVC asymétriques de 325 villes et des PVC symétriques de 120 villes ont été résolus de façon exacte. Nous décrirons plus tard les algorithmes qui ont atteint ces tailles.

La littérature concernant le PVC est très abondante. Notre résumé en sera volontairement incomplet. Nous nous concentrerons sur:

1. les principaux modèles du PVC;
2. les algorithmes optimaux les plus efficaces pour le résoudre.

Deux critères nous servent principalement pour évaluer la performance d'un algorithme optimal résolvant un problème de type PEV:

1. l'ordre de grandeur de la taille (habituellement en termes du nombre de villes) maximum des problèmes qu'il a résolus;
2. son temps d'exécution moyen pour les différentes tailles des problèmes résolus.

Le plan de ce chapitre est le suivant:

- section 2.2 : modèles pour les PVC asymétriques;
- section 2.3 : algorithmes optimaux pour les PVC asymétriques;
- section 2.4 : modèles pour les PVC symétriques;
- section 2.5 : algorithmes optimaux pour les PVC symétriques;

---

(1) Coupes de Gomory, "Branch-and-Bound", relaxation lagrangienne, etc.

- section 2.6 : résolution des PVC symétriques par l'utilisation de la programmation linéaire en nombres entiers;
- section 2.7 : conclusion.

Trois remarques s'imposent au sujet de ce plan:

1. aucun heuristique pour le PVC ne sera décrit dans ce chapitre[1];
2. les PVC asymétriques sont clairement distingués des PVC symétriques dans ce plan;
3. la section 2.5 aurait pu contenir le matériel de la section 2.6; toutefois, les algorithmes présentés dans cette dernière section sont à la base de nos travaux et nous avons voulu les regrouper dans une même section.

2.2 Formulations des PVC asymétriques

2.2.1 Introduction

Dans la section 2.2 on suppose que la matrice C des coûts de transport est asymétrique. Le graphe G = (N,A) d'ordre n associé au PVC est donc orienté et la résolution du PVC constitue la recherche du circuit hamiltonien (ou tour)[2] de coût minimal. Soit G' = (N,A') un sous-graphe de G formant un tour dans G. Le graphe G' possède les deux propriétés suivantes:

1. tous les sommets de G' ont un degré intrant et un degré extrant de 1;
2. G' est connexe.

La majorité des modèles mis au point pour le PVC asymétrique possèdent la forme suivante:

---

(1) [51] contient une bibliographie de tels heuristiques.

(2) Lorsqu'un seul commis est utilisé, un tour est un cycle ou circuit hamiltonien de G.

(M1)

$$\text{Min } Z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \quad (1) \qquad (2.2)$$

assujettie à:

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad (i = 1,\ldots,n) \qquad (2.3)$$

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad (j = 1,\ldots,n) \qquad (2.4)$$

Les x_{ij} ne forment pas de sous-tours (2.5)

$$x_{ij} = 0,1 \qquad (i,j = 1,\ldots,n) \qquad (2.6)$$

La signification des variables x_{ij} a été donnée au premier chapitre. On compte n(n-1) telles variables. Dans (2.2) le coût du circuit hamiltonien est minimisé. Les contraintes (2.3) et (2.4) sont associées à la propriété 1 de G'. Elles sont appelées les contraintes de degré. Le modèle M1 sans (2.5) est le modèle du problème d'affectation [13] abrégé par PA tout au long de ce texte. Conséquemment dans le cas asymétrique les contraintes de degré seront aussi appelées les contraintes d'affectation.

Les contraintes d'affectation ne suffisent pas à elles seules à assurer que G' soit connexe. Le graphe correspondant à une solution réalisable de PA constitue soit un tour, soit un ensemble de plus d'un sous-tour. Les contraintes de (2.5) servent à empêcher l'occurence de ce dernier type de solutions, tout en étant satisfaites par toute solution de PA représentant un tour. Elles sont appelées les contraintes d'élimination de sous-tours. Dans les 3 sections qui suivent nous expliciterons 3 expressions mathématiques différentes des contraintes de (2.5) conduisant à 3 modèles différents pour le PVC asymétrique.

Nous avons retrouvé dans la littérature un modèle pour le PVC

---

(1) Dans toute somme sur les variables x_{ij}, il sera implicite que $i \neq j$.

asymétrique qui diffère grandement de M1. Il est dû à Svestka [118] et est basé sur la notion de contrainte de cardinalité. A notre connaissance aucun algorithme jusqu'à présent ne peut traiter aisément ce type de contrainte et le lecteur est référé à [118] pour plus de détails à ce sujet.

2.2.2 Contraintes d'élimination de sous-tours de Dantzig et al. [28]

Dantzig et al. [28] en 1954, dans leur article intitulé "Solution of a large scale[1] travelling salesman problem" furent, à notre connaissance, les premiers à mettre sur pied des contraintes d'élimination de sous-tours pour le PVC. Elles ont été originalement conçues par leurs auteurs pour traiter d'un PVC symétrique. L'extension au cas asymétrique des contraintes de Dantzig et al. est triviale à effectuer et les contraintes d'élimination de sous-tours qui suivent leur sont habituellement attribuées:

$$\sum_{i\in S, j\in \bar{S}} x_{ij} \geq 1 \qquad (S \subset N) \qquad (2.7)$$

où $\bar{S} = N-S$

Considérons une solution de PA dont le graphe associé contient le sous-tour $(i_1,\ldots,i_k,i_1)$ où $2 \leq k \leq n-2$[2]. Si $S = \{i_1,\ldots,i_k\}$, (2.7) est clairement violée. Par contre, si le graphe de la solution de PA considérée représente un tour, ce graphe est connexe et pour toute partition $(S,\bar{S})$ de N (2.7) est satisfaite.

Dans (2.7) la cardinalité de S peut varier de 2 jusqu'à (n-2). Le nombre de contraintes qu'on y retrouve, dénoté NB, peut donc s'écrire:

$$NB = \begin{cases} 2\left[\binom{n}{2} + \ldots + \binom{n}{\frac{1}{2}(n-1)}\right] & \text{si n impair} \\ 2\left[\binom{n}{2} + \ldots + \binom{n}{\frac{n}{2}-1}\right] + \binom{n}{\frac{n}{2}} & \text{si n pair} \end{cases} \qquad (2.8)$$

---

(1) Ce problème contient 42 villes.

(2) Des sous-tours contenant (n-1) villes ne peuvent se présenter lorsque les contraintes d'affectation sont satisfaites.

NB est approximativement égal à 2^n et il est clair que l'utilisation de (2.7) dans le modèle M1 rend ce dernier impossible à résoudre directement[1], et cela même si n est relativement petit.

Dantzig et al. [28] ont démontré que les contraintes suivantes sont équivalentes aux contraintes de (2.7):

$$\sum_{i\in S, j\in S} x_{ij} \leq |S|-1 \qquad (S \subset N) \qquad (2.9)$$

Nous avons mis au point le théorème (2.1) qui suit de façon à:

1. généraliser l'équivalence entre (2.7) et (2.9);
2. utiliser ce théorème pour prouver certains nouveaux résultats qui seront présentés aux chapitres 4 et 6.

Théorème 2.1

Soit $S \subset N$ un ensemble de sommets satisfaisant aux égalités suivantes:

$$\sum_{j=1}^{n} x_{ij} = \sum_{j=1}^{n} x_{ji} = 1 \qquad (i \in S) \qquad (2.10)$$

Soit τ un nombre réel et $\bar{S} = N-S$.

L'équivalence suivante est valide:

$$\sum_{i\in S, j\in \bar{S}} x_{ij} \geq \tau \Leftrightarrow \sum_{i\in S, j\in S} x_{ij} \leq |S|-\tau \qquad (2.11)$$

Preuve:

On peut écrire:

$$\sum_{i\in S, j\in \bar{S}} x_{ij} \geq \tau \Leftrightarrow \sum_{i\in S} \left[\sum_{j\in \bar{S}} x_{ij}\right] \geq \tau \Leftrightarrow \sum_{i\in S} \left[1 - \sum_{j\in S} x_{ij}\right] \geq \tau$$

---

(1) C'est-à-dire en considérant explicitement dans la formulation toutes les contraintes de M1.

la dernière équivalence étant due à (2.10). La dernière égalité s'écrit aussi:

$$|S| - \sum_{i\in S, j\in S} x_{ij} \geq \tau$$

On obtient donc:

$$\sum_{i\in S, j\in \bar{S}} x_{ij} \geq \tau \quad \Leftrightarrow \quad |S| - \sum_{i\in S, j\in S} x_{ij} \geq \tau$$

ce qui est une réécriture de (2.11). ■

Lorsque $\tau = 1$, (2.11) établit l'équivalence entre (2.7) et (2.9).

D'autres contraintes d'élimination de sous-tours, très étroitement liées à celles de Dantzig et al. peuvent aussi être utilisées dans M1. Soit S' l'ensemble des arcs d'un sous-tour dans $S \subset N$, la contrainte suivante est valide pour le PVC:

$$\sum_{(i,j)\in S'} x_{ij} \leq |S| - 1 \qquad (2.12)$$

(2.12) n'élimine qu'un sous-tour précis pouvant se former dans S tandis que (2.9) élimine tout sous-tour de taille |S| dans S. (2.12) est donc moins forte[1] que la contrainte correspondante dans (2.9).

2.2.3 Contraintes d'élimination de sous-tours de Miller et al. [94]

Miller et al. [94] en 1960 découvrirent un autre type de contraintes d'élimination de sous-tours pour le PVC asymétrique. Leurs contraintes peuvent s'écrire:

$$u_i - u_j + n\, x_{ij} \leq n-1 \qquad (2 \leq i \neq j \leq n) \quad (2.13)$$

où $u_i (i = 2,\ldots,n)$ est un réel quelconque.

---

(1) Une contrainte $a^0 \cdot x \leq b^0$ est moins forte qu'une autre contrainte $a^1 \cdot x \leq b^1$ si celle-ci implique l'autre mais pas nécessairement l'inverse.

(2.13) empêche la formation de tout sous-tour ne contenant pas la ville 1, c'est-à-dire le dépôt, et par le fait même la formation de tout sous-tour[1]. Pour s'en convaincre considérons le sous-tour $t=(i_1,\ldots,i_k,i_1)$ où $2 \leq k \leq n-2$ et où $i_j \neq 1$ pour $j = 1,\ldots,k$; considérons aussi les contraintes suivantes de (2.13):

$$\begin{bmatrix} u_{i_1} - u_{i_2} + nx_{i_1,i_2} \leq n-1 \\ u_{i_2} - u_{i_3} + nx_{i_2,i_3} \leq n-1 \\ \vdots \\ u_{i_{k-1}} - u_{i_k} + nx_{i_{k-1},i_k} \leq n-1 \\ u_{i_k} - u_{i_1} + nx_{i_k,i_1} \leq n-1 \end{bmatrix} \qquad (2.14)$$

Si le sous-tour t apparaît dans la solution de PA on aura alors $x_{i_j,i_{j+1}} = 1$ $(j = 1,\ldots,k-1)$ et $x_{i_k,i_1} = 1$. En additionnant les contraintes de (2.14) on obtient:

$$kn \leq k(n-1) \qquad (2.15)$$

re qui est une contradiction.

Pour un PVC asymétrique avec n villes on compte (n-1)(n-2) contraintes dans (2.13) et (n-1) variables continues sont ajoutées à M1. Le nombre des contraintes de Miller et al. croît donc de façon polynomiale en terme de n comparativement à une croissance exponentielle pour celui des contraintes de Dantzig et al. On peut démontrer toutefois que ces dernières contraintes sont plus "fortes" que celles de Miller et al. dans le sens suivant. Additionnons de nouveau les contraintes de (2.14) mais sans spécifier de valeur précise aux variables impliquées dans ces contraintes. On obtient:

$$\sum_{(i,j)\in S'}^{n} x_{ij} \leq kn-k \qquad (2.16)$$

---

(1) Par un raisonnement similaire on pourrait considérer $S \subseteq N-\{1\}$ au lieu de $S \subset N$ dans (2.7) et (2.9).

où S' est l'ensemble des arcs du sous-tour t précédent. On peut réécrire (2.16) ainsi:

$$\sum_{(i,j)\in S'} x_{ij} \leq |S| - \frac{|S|}{n} \qquad (2.17)$$

$$\text{où } S = (i_1,\ldots,i_k)$$

Puisque $|S| < n$, (2.17) est moins forte que

$$\sum_{(i,j)\in S'} x_{ij} \leq |S| - 1 \qquad (2.18) \equiv (2.12)$$

ce qui est, comme on l'a vu, une contrainte moins forte que (2.9). En fait, tout comme (2.12), les contraintes de (2.14) n'éliminent qu'un seul sous-tour pouvant se former sur les sommets de S, ce qui n'est pas le cas des contraintes de Dantzig et al.

2.2.4 Contraintes d'élimination de sous-tours de Gavish et Graves [45]

Récemment, en 1979, Gavish et Graves [45] ont établi deux nouvelles formulations mathématiques pour le PVC asymétrique. Elles sont toutes deux basées sur l'introduction de variables de flot dans M1 mais elles se distinguent par les contraintes d'élimination de sous-tours utilisées. Nous n'exposerons ici qu'un de ces deux modèles, l'autre étant assez similaire.

Soit y_{ij} $(i \neq j)$ une variable de flot non-négative associée à l'arc (i,j) du graphe G correspondant au PVC. Gavish et Graves ont démontré que les contraintes suivantes empêchent la formation de sous-tours:

$$\sum_{j=1}^{n} y_{ji} - \sum_{j=2}^{n} y_{ij} = 1 \qquad (i = 2,\ldots,n) \qquad (2.19)$$

$$y_{ij} \leq (n-1)x_{ij} \qquad (i=1,\ldots,n;\ j=2,\ldots,n) \qquad (2.20)$$

Pour expliquer le fonctionnement de (2.19) et (2.20), considérons une solution de PA contenant le sous-tour $(i_1,\ldots,i_k,i_1)$ où $2 \leq k \leq n-2$ et où $i_j \neq 1$ pour $j = 1,\ldots,k$. Supposons sans perte de généralité que y_{i_k,i_1} prend une valeur quelconque f non-négative. Par (2.19) et (2.20) il s'ensuit que $y_{i_j,i_{j+1}} = f+k-j$ pour $j = 1,\ldots,k-1$. On obtient donc:

$$\sum_{j=1}^{n} y_{j,i_1} - \sum_{j=2}^{n} y_{i_1,j} = f - (f+k-1) = 1-k \qquad (2.21)$$

et (2.21) contredit (2.19).

La contrainte (2.19) stipule le flot entrant dans un sommet $i \neq 1$ moins le flot sortant de i doit égaler 1. A chaque sommet $i \neq 1$ il doit donc se produire une perte d'une unité de flot et ceci est clairement impossible dans un sous-tour ne contenant pas le sommet 1. Les seules solutions réalisables de PA satisfaisant aussi à (2.19) et à (2.20) constitueront donc des tours et l'attribution de valeurs aux variables y_{ij} pour un tour donné est simple: l'arc incident vers l'extérieur au sommet 1, soit l'arc (1,j), reçoit un flot de $y_{1j} = (n-1)$; l'arc incident à j vers l'extérieur reçoit un flot de (n-2) unités, etc. jusqu'à ce que le tour soit complété. En procédant ainsi (2.19) et (2.20) seront clairement satisfaites. Notons que pour des valeurs fixées des variables x_{ij} ces contraintes définissent un problème de flot. Il s'ensuit alors que les variables y_{ij} prendront des valeurs entières.

L'utilisation des contraintes de Gavish et Graves comme contraintes d'élimination de sous-tours implique l'introduction dans M1 de $n(n-1)$ contraintes et de $(n-1)^2$ variables de flot. Tout comme les contraintes de Miller et al., celles de Gavish et Graves sont donc en nombre polynomial en terme de n et beaucoup moins nombreuses que les contraintes de Dantzig et al.

Gavish et Graves [45] ont démontré que:

1. il existe un certain lien dual entre leur formulation du PVC et celle de Miller et al. (voir [45]);

2. par l'application de la décomposition de Benders [9] sur leur formulation du PVC, on retrouve les contraintes de Dantzig et al. pour ce même problème(1).

2.3 Algorithmes optimaux pour les PVC asymétriques

2.3.1 Introduction

Le PVC fait partie de la classe des problèmes NP-complets et conséquemment la majorité des algorithmes optimaux pour le résoudre sont du type "Branch-and-Bound". La construction d'un tel algorithme exige le calcul d'une borne inférieure(2) obtenue à partir d'une relaxation du problème à résoudre. Trois critères principaux influencent l'efficacité d'une méthode de "Branch-and-Bound" basée sur une borne donnée:

1. la rapidité avec laquelle on peut calculer cette borne;
2. l'efficacité avec laquelle on l'utilise dans l'arbre de l'énumération, c'est-à-dire le critère de branchement employé;
3. la qualité de cette borne, c'est-à-dire sa proximité par rapport à la solution optimale du problème à résoudre.

Le problème d'affectation est une relaxation du PVC et V(PA) constitue une borne inférieure excellente sur V(PVC) selon les 3 critères précédents. En effet:

---

(1) Dans le chapitre 4 de ce texte nous démontrerons un résultat similaire au sujet du c.m-PVC.

(2) Borne supérieure dans le cas d'une maximisation.

1. le calcul de V(PA) peut s'effectuer en $O(n^3)$ opérations par plusieurs méthodes différentes dont [31,122];
2. selon Christofides [17], V(PA) est, en moyenne, seulement 3% inférieure à V(PVC);
3. PA est unimodulaire et les variables x_{ij} prendront automatiquement une valeur entière dans toute solution optimale.

Dans la section 2.3.2 les algorithmes pour le PVC asymétrique basés sur V(PA) seront décrits brièvement. La section 2.3.3 contiendra quelques résultats concernant l'utilisation de bornes autres que V(PA) pour la résolution de ce problème.

2.3.2 Méthodes de "Branch-and-Bound" pour le PVC asymétrique basées sur le problème d'affectation

Plusieurs algorithmes sont apparus dans la littérature où le PVC asymétrique est résolu par une méthode de "Branch-and-Bound" utilisant le problème d'affectation (PA) dans le calcul d'une borne inférieure. Nous décrirons premièrement de façon générale le modus operandi commun à la plupart de ces algorithmes pour ensuite montrer en quoi ils se distinguent les uns des autres.

Les notations suivantes nous seront utiles:

P^c le noeud de l'arbre présentement sous considération

PVC^c le PVC associé au noeud P^c

PA^c le problème d'affectation associé à PVC^c

$\underline{Z}$ une borne inférieure sur $V(PVC^c)$ obtenue par la résolution de PA^c ($\underline{Z} = V(PA^c)$)

$X(\underline{Z})$ une solution optimale de PA^c (de coût $\underline{Z}$)

Z^* le coût du meilleur tour connu (initialement on peut poser $Z^* = \infty$).

A chaque noeud P^c de l'arbre correspond le problème PVC^c qui diffère uniquement du PVC original à résoudre par le fait que certaines variables x_{ij} sont fixées à la valeur "0" ou bien à la valeur "1". Au noeud P^c, le problème PVC^c n'est pas résolu mais on calcule plutôt une borne inférieure sur $V(PVC^c)$ par la résolution de façon exacte de PA^c. Trois situations peuvent se produire à ce noeud une fois le calcul de $\underline{Z}$ effectué:

1. $\underline{Z} \geq Z^*$
2. $\underline{Z} < Z^*$ et $X(\underline{Z})$ est un tour
3. $\underline{Z} < Z^*$ mais $X(\underline{Z})$ contient des sous-tours

Dans les situations 1 et 2 le noeud P^c peut être éliminé des considérations et il faut choisir un nouveau noeud à analyser. En plus, dans la situation 2, on peut poser $Z^* = \underline{Z}$.

La situation 3 est plus complexe à traiter. Puisque $\underline{Z} < Z^*$, $F(PVC^c)$ peut contenir la solution optimale du PVC original mais $X(\underline{Z}) \notin F(PVC^c)$ car cette solution contient des sous-tours. A partir de P^c on créera donc k nouveaux noeuds, notés $P^c_1,\ldots,P^c_k$, où[1]:

1. $X(\underline{Z}) \notin F(PA^c_i) \qquad (i = 1,\ldots,k);$
2. $\bigcup_{i=1}^{k} F(PVC^c_i) = F(PVC^c).$

Pour s'assurer que la condition 1 précédente soit satisfaite, un sous-tour de $X(\underline{Z})$ est sélectionné et on s'assure, par la fixation de certaines variables[2] à la valeur 0 ou 1, qu'aucune solution réalisable des problèmes PA^c_i ne pourra le contenir.

Quelle que soit la situation dans laquelle on se retrouve au noeud P^c, il faudra sélectionner un nouveau noeud à considérer[3]. On rencontre

---

(1) Les problèmes du type PA et PVC associés à ces nouveaux noeuds sont dénotés PA^c_i et PVC^c_i pour $i = 1,\ldots,k$.

(2) Ce qui s'effectue par une simple modification des coûts des variables correspondantes i.e. $x_{ij} = 1$ (0) $\Rightarrow$ $c_{ij} = -\infty$ ($+\infty$).

(3) Dans la situation 3, la liste des noeuds candidats (pendants) est augmentée de $P^c_1,\ldots,P^c_k$ avant cette sélection.

principalement deux méthodes de branchement:

1. sélection du noeud pendant possédant la plus petite borne inférieure (méthode de branchement 1);
2. sélection du noeud pendant possédant la plus petite borne inférieure parmi les noeuds pendants au niveau le plus élevé (méthode de branchement 2).

Le but de la méthode de branchement 1 précédente est de trouver rapidement une bonne valeur de Z^* et pouvoir ainsi éliminer plusieurs noeuds pendants. La gestion de l'arbre est toutefois en général plus lourde[1] à effectuer que la méthode de branchement 2 (du type LIFO). La désavantage de cette dernière réside dans le fait que le niveau de l'arbre peut devenir rapidement élevé et son avantage se situe dans la possibilité de pouvoir trouver rapidement $V(PA_i^c)$ à partir de la solution optimale de PA^c, principalement à l'aide de techniques de paramétrisation.

Nous avons relevé dans la littérature les méthodes suivantes pour résoudre le PVC asymétrique de la façon décrite précédemment:

1. Shapiro, 1966, [111];
2. Bellmore et Malone, 1971, [7];
3. Smith et al., 1977, [114];
4. Carpaneto et Toth, 1980, [15].

Algorithmes de Shapiro [111]

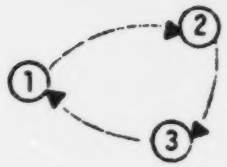
. Méthode de branchement 1;

. Création de $P_1^c,\ldots,P_k^c$ à partir de P^c par l'élimination du sous-tour contenant le plus petit nombre de villes[2] i.e. supposons que la solution optimale de PA^c possède comme plus petit sous-tour:

---

(1) Surtout parce que le nouveau noeud pendant choisi peut être à un niveau de beaucoup inférieur au noeud sous considération.

(2) On dira avec un certain abus de langage "le plus petit sous-tour".



A partir de P^c les noeuds P_1^c, P_2^c et P_3^c sont créés où dans P_1^c on fixe x_{12} à 0, dans P_2^c on fixe x_{23} à 0 et dans P_3^c on fixe x_{31} à 0.

. Procédure de création de noeuds basée sur

$$\sum_{(i,j)\in S'} x_{ij} \leq |S|-1 \quad (1)$$

où S' est l'ensemble des arcs du plus petit sous-tour sur S;

. Tests sur des PVC avec matrice C asymétrique et symétrique(2);

. Temps moyen de 43 secondes sur 10 PVC de 40 villes. (IBM 360-65).

Algorithmes de Bellmore et Malone [7]

. Méthode de branchement 1;

. Résolution de PA par l'algorithme d'Hitchcock [63];

. Choix du plus petit sous-tour pour la création de noeuds mais procédure différente de celle de Shapiro i.e. supposons que le plus petit sous-tour dans la solution optimale de PA^c associé à P^c soit:

---

(1) C'est la contrainte (2.12) vue en 2.2.2. Au moins une des variables x_{ij} où $(i,j) \in S'$ doit être égale à 0.

(2) Les PVC symétriques sont considérés comme des PVC asymétriques.

Les noeuds P_1^c, P_2^c et P_3^c sont créés à partir de P^c ainsi:

dans PA_1^c : $x_{1j} = 0$ $\quad (j \in \{1,2,3\})$

dans PA_2^c : $x_{2j} = 0$ $\quad (j \in \{1,2,3\})$

dans PA_3^c : $x_{3j} = 0$ $\quad (j \in \{1,2,3\})$

. Procédure de création de noeuds basée sur

$$\sum_{i \in S, j \in \bar{S}} x_{ij} \geq 1 \quad (1)$$

où S est l'ensemble des sommets du plus petit sous-tour;

. Bellmore et Malone ont démontré que cette procédure de création de noeuds élimine un plus grand nombre de solutions réalisables pour PA^c et irréalisable pour PVC^c que la procédure de Shapiro(2);

. La procédure de la création de noeuds de Bellmore et Malone n'engendre pas de PA_j^c tels que leurs domaines réalisables sont mutuellement disjoints;

Temps moyen de 18 secondes pour 10 PVC de 40 villes (IBM 360-65);

. PVC de taille n = 80 résolus en un temps moyen de 165 secondes (IBM 360-65).

Algorithme de Smith et al. [114]

. Méthode de branchement 2;

. Procédure de création de noeuds similaire à celle de Bellmore et Malone;

---

(1) C'est la contrainte (2.7) vue en 2.2.2. En posant $x_{ij} = 0$ pour tout $j \in \{1,2,3\}$, au moins un arc incident à i vers l'extérieur sera dirigé vers S.

(2) C'est un résultat assez évident car on a vu que (2.7) est beaucoup plus forte que (2.12).

. Solution très rapide de PA_j^c par paramétrisation à partir de la solution optimale de PA^c (Smith et al. ont développé une théorie des opérateurs à cette fin).

. PVC de taille n = 180 résolus en un temps moyen de 430 secondes (UNIVAC 1108);

. Trois versions différentes ont été testées.

Algorithme de Carpaneto et Toth [15]

. Méthode de branchement 2;

Raffinements apportés à l'algorithme de Smith et al.:

- choix, non pas du plus petit sous-tour, mais de celui contenant le plus petit nombre d'arcs dont les variables associées ne sont pas fixées à la valeur 1;
- addition de contraintes supplémentaires à chaque noeud de l'arbre;
- calcul de bornes inférieures partielles;
- résolution par paramétrisation des PA engendrés;

PVC de taille n = 240 résolus en un temps moyen de 6.5 secondes (CDC 6600).

Plus récemment, en 1981, Balas et Christofides [2] ont amélioré la qualité de la borne inférieure obtenue sur V(PVC) en utilisant le problème d'affectation mais en ajoutant de façon lagrangienne certaines contraintes de (2.7) et (2.12) dans l'objectif du PA. L'introduction de quelques contraintes de (2.7) dans l'objectif conduit à la résolution du problème suivant:

$$(PA(\lambda)) \quad \left[\begin{array}{l} \text{Min } Z = \{ \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} - [\sum_{t} \lambda_t (\sum_{\substack{i \in S_t \\ j \in \bar{S}_t}} x_{ij})] + \sum_{t} \lambda_t \} \\ \\ \underline{\text{assujettie à}}: \\ \\ (2.3),\ (2.4) \text{ et } (2.6). \end{array} \right.$$

Dans PA(λ), λ_t est le multiplicateur de Lagrange associé à la contrainte t de (2.7) et λ est le vecteur des λ_t. Evidemment le nombre de λ_t est exponentiel et seulement un sous-ensemble restreint d'entre eux seront considérés dans PA(λ).

La recherche de $\bar{\lambda}$ tel que $PA(\bar{\lambda}) = \max_{\lambda} \{PA(\lambda)\}$ permet à Balas et Christofides de calculer une borne inférieure sur V(PVC) qui est seulement 1% inférieure en moyenne à V(PVC).

L'utilisation de cette borne dans une méthode de "Branch-and-Bound" a permis de résoudre optimalement des PVC de taille n = 325 en un temps moyen de 50 secondes (CDC 7600).

Finalement, nous désirons souligner que l'un des premiers algorithmes pour le PVC est dû à Little et al. [85]. Cet algorithme utilise aussi le problème d'affectation dans le calcul d'une borne inférieure sur V(PVC). Il est toutefois assez différent de la méthode générale de "Branch-and-Bound" pour le PVC décrite précédemment. De plus son efficacité est largement dépassée par les plus récents algorithmes mis au point pour ce problème et c'est pourquoi il n'a pas été décrit dans ce chapitre.

2.3.3 Bornes inférieures différentes de V(PA) sur V(PVC)

Quelques bornes inférieures sur V(PVC), distinctes de V(PA), ont été analysées par les chercheurs. Elles sont généralement obtenues par la relaxation, non pas des contraintes d'élimination de sous-tours, mais bien de celles des contraintes d'affectation. Mentionnons entre autres:

1. la borne obtenue par le calcul de n-chemins de coûts minimaux [65];

2. la borne obtenue par le calcul d'une 1-arborescence de poids minimal[1] [113].

---

(1) Contrepartie asymétrique d'un 1-arbre qui sera décrit en 2.5.

Les méthodes de "Branch-and-Bound" associées à ces bornes inférieures se sont avérées toutefois moins efficaces que les algorithmes de Carpaneto et Toth [15] et que celui de Balas et Christofides [2]. Elles ne seront pas décrites dans ce texte.

2.4 Formulations des PVC symétriques

Dans la section 2.4 la matrice des coûts de transport est supposée symétrique. Le graphe G = (N,A) d'ordre n correspondant au PVC est non-orienté. Soit G' = (N,A') un sous-graphe de G représentant un cycle hamiltonien dans G. Le graphe G' satisfait aux deux conditions suivantes:

1. tous les sommets de G' possèdent un degré de 2;
2. G' est connexe.

La résolution du PVC symétrique implique la recherche, parmi tous les sous-graphes de G possédant ces deux caractéristiques, de celui ayant un poids minimal. En associant une variable x_{ij} où $i < j$ à chaque arête du graphe G, Dantzig et al. [28] ont élaboré le modèle suivant pour le PVC symétrique:

(M2)

$$\text{Min } Z = \sum_{i<j} c_{ij} x_{ij} \tag{2.22}$$

assujettie à:

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2 \qquad (k = 1,\ldots,n) \tag{2.23}$$

$$\sum_{\substack{i\in S, j\in S \\ i<j}} x_{ij} \leq |S|-1 \qquad (S \subset N) \tag{2.24}$$

$$x_{ij} = 0,1 \qquad (1 \leq i<j \leq n) \tag{2.25}$$

Le modèle M2 s'interprète de façon tout à fait similaire au modèle M1 de la section 2.2.1. L'interprétation des variables x_{ij}, au nombre de $\frac{1}{2}n(n-1)$, a été vue au premier chapitre. Dans (2.22) le coût du cycle hamiltonien est minimisé. Les contraintes (2.23), dénommées les contraintes de degré, correspondent à la condition 1 précédente. Comme dans le cas asymétrique le graphe d'une solution réalisable de M2, sans les contraintes (2.24), n'est pas nécessairement connexe. Ces dernières sont appelées encore ici les contraintes d'élimination de sous-tours et visent à assurer que la condition 2 précédente soit satisfaite. Le modus operandi des contraintes (2.24) est analogue à leur contrepartie asymétrique, notamment (2.9). Notons toutefois que les sous-tours contenant 2 villes sont implicitement éliminés par le choix des variables x_{ij} où $i < j$[1]. La cardinalité de S dans (2.24) peut donc varier de 3 à n-3.

Dantzig et al. ont démontré que les contraintes suivantes sont équivalentes à (2.24):

$$\sum_{\substack{i\in S, j\in \bar{S} \\ \text{ou} \\ i\in \bar{S}, j\in S \\ i<j}} x_{ij} \geq 2 \qquad (S \subset N) \qquad (2.26)$$

où $\bar{S} = N-S$

Cette équivalence découle directement du théorème suivant. Nous l'avons construit de façon plus générale car son résultat nous sera utile dans les prochains chapitres.

Théorème 2.2

Soit $S \subset N$ un ensemble de sommets satisfaisant à l'égalité suivante:

---

(1) Pour modéliser un tel sous-tour contenant les villes i et j, il faudrait permettre à la variable x_{ij} de prendre la valeur 2.

$$\sum_{i<k} x_{ik} + \sum_{i>k} x_{kj} = 2 \qquad (k \in S) \qquad (2.27)$$

Soit τ un nombre réel et $\bar{S} = N-S$.

L'équivalence suivante est valide:

$$\sum_{i\in S, j\in S \atop i<j} x_{ij} \leq |S|-\tau \quad \Leftrightarrow \quad \sum_{i\in S, j\in \bar{S} \atop \text{ou} \atop i\in \bar{S}, j\in S \atop i<j} x_{ij} \geq 2\tau \qquad (2.28)$$

Preuve:

Par (2.27) on a:

$$\sum_{i\in S} \left[\sum_{j>i} x_{ij} + \sum_{k<i} x_{ki} \right] = 2 \cdot |S|$$

ce qui, après manipulations peut s'écrire:

$$\sum_{i\in S} \sum_{\substack{j\in S \\ j>i}} x_{ij} + \sum_{i\in S} \sum_{\substack{k\in S \\ k<i}} x_{ki} = 2 \cdot |S| - \sum_{i\in S} \sum_{\substack{j\in \bar{S} \\ j>i}} x_{ij} - \sum_{i\in S} \sum_{\substack{k\in \bar{S} \\ k<i}} x_{ki}$$

On obtient donc:

$$2 \cdot \sum_{i\in S, j\in S \atop i<j} x_{ij} = 2 \cdot |S| - \sum_{i\in S, j\in \bar{S} \atop \text{ou} \atop i\in \bar{S}, j\in S \atop i<j} x_{ij}$$

et donc:

$$\sum_{i\in S, j\in S \atop i<j} x_{ij} = |S| - \tfrac{1}{2} \cdot \sum_{i\in S, j\in \bar{S} \atop \text{ou} \atop i\in \bar{S}, j\in S \atop i<j} x_{ij}$$

De cette dernière égalité on déduit que:

$$\sum_{\substack{i\in S, j\in S \\ i<j}} x_{ij} \leq |S| - \tau = |S| - \tfrac{1}{2} \cdot \sum_{\substack{i\in S, j\in \bar{S} \\ \text{ou} \\ i\in \bar{S}, j\in S \\ i<j}} x_{ij} = |S| -$$

ce qui nous permet d'obtenir (2.28). ■

En prenant $\tau = 1$, l'équivalence entre (2.24) et (2.26) est démontrée. Le nombre de contraintes d'élimination de sous-tours dans M2 peut donc être évalué par le nombre de partitions différentes $(S,\bar{S})$ de N, c'est-à-dire:

$$\binom{n}{3} + \ldots + \binom{n}{\left\lfloor \frac{n}{2} \right\rfloor} \qquad (2.29)$$

ce qui est exponentiel en n.

Dans (2.29), le terme $\binom{n}{2}$ n'a pas été inclus car les sous-tours contenant deux villes ne peuvent se produire.

Notons que les contraintes d'élimination de sous-tours (2.13) de Miller et al. ainsi que celles de Gavish et Graves, (2.19-20), ne peuvent s'adapter au cas symétrique contrairement à celles de Dantzig et al.

2.5 Algorithmes optimaux pour les PVC symétriques

Puisque tout PVC symétrique peut se formuler comme un PVC asymétrique en utilisant conjointement des variables x_{ij} et x_{ji} de même coût, les méthodes exposées en 2.3 pour le PVC asymétrique peuvent aussi servir dans le cas symétrique. Toutefois leur performance est grandement affectée par la symétrie dans C car, selon Christofides [17], $\left(\frac{V(PVC)-V(PA)}{V(PVC)}\right)$ vaut alors en moyenne 0.2(1), la symétrie dans les coûts occasionnant la fréquente occurence de sous-tours de 2 ou 3 villes(2).

---

(1) Comparativement à 0.03 dans le cas asymétrique.

(2) Principalement dans le cas symétrique euclidien.

Bellmore et Malone [7] furent, à notre avis, les premiers à démontrer clairement à l'aide de tests exhaustifs qu'il fallait séparer le cas asymétrique du cas symétrique. Ils élaborèrent dans ce même article un algorithme pour le cas symétrique, similaire à leur algorithme pour le cas asymétrique décrit en 2.3.2,où la relaxation utilisée à chaque noeud de l'arbre de l'énumération est le modèle M2 sans (2.24). C'est un problème de "2-matching" pouvant être résolu par l'algorithme polynomial d'Edmonds [33] en au plus n^4 opérations.

Cet algorithme de Bellmore et Malone s'est avéré toutefois très peu efficace, prenant un temps moyen de 357 secondes (UNIVAC 1108) sur 10 PVC symétriques de taille n = 20. Les raisons probables de cette piètre performance sont:

1. la présence de nombreux sous-tours contenant 3 villes dans la solution optimale du problème de "2-matching"(1);
2. le fait que le temps de résolution d'un problème de "2-matching" est plus long en général que celui nécessaire à la résolution du problème d'affectation.

Held et Karp [60,61] ont élaboré en 1970-71 une des méthodes exactes de résolution du PVC symétrique qui compte encore aujourd'hui parmi les plus efficaces pour ce problème. Leur algorithme est basé sur la notion d'un 1-arbre défini comme un arbre sur l'ensemble de sommets {2,...,n} plus deux arêtes incidentes au sommet 1. Puisque tout cycle hamiltonien est un 1-arbre et que l'inverse n'est pas nécessairement vrai, le problème PT suivant constitue une relaxation du PVC:

$$(PT) \quad \left[\begin{array}{ll} \text{Min } Z = \sum_{i<j} c_{ij} x_{ij} & (2.30) \\ \\ \underline{\text{assujettie à}}: & \\ \\ x = \{x_{ij}\} \text{ forme un 1-arbre} & (2.31) \end{array} \right.$$

(1) Phénomène observé par Bellmore et Malone.

De ce fait V(PT) est une borne inférieure valide sur V(PVC). Elle est obtenue en relaxant les contraintes de degré dans M2. Le calcul de V(PT) peut s'effectuer très rapidement par un algorithme polynomial [68].

En modifiant la matrice des coûts C, V(PT) conduit à toute une famille de bornes sur V(PVC). Les trois constatations suivantes nous permettent de voir comment:

1. une solution optimale du PVC avec matrice de coûts $C = (c_{ij})$ est aussi une solution optimale du PVC avec matrice de coûts $C' = (c'_{ij})$ où

$$c'_{ij} = c_{ij} + \pi_i + \pi_j \tag{2.32}$$

et où $\pi = (\pi_1, \ldots, \pi_n)$ est un vecteur de nombres réels;

2. si c^* est le coût optimal du PVC sur $C = (c_{ij})$ alors $c^* + 2 \sum_{i=1}^{n} \pi_i$ est le coût optimal du PVC sur $C' = (c'_{ij})$;

3. le 1-arbre de poids minimal avec $C = (c_{ij})$ comme matrice de poids n'est pas nécessairement identique au 1-arbre de poids minimal avec C' définissant les poids.

Soit PT_π le problème suivant:

$$(PT_\pi) \quad \begin{cases} \text{Min } Z = \sum_{i<j} (c_{ij} + \pi_i + \pi_j) x_{ij} & (2.33) \\ \underline{\text{assujettie à}}: \\ x = \{x_{ij}\} \text{ forme un 1-arbre} & (2.34) \end{cases}$$

Par les constatations précédentes on voit clairement que:

$$V(PT_\pi) \leq c^* + 2 \sum_{i=1}^{n} \pi_i \tag{2.35}$$

et donc:

$$V(PT_\pi) - 2 \sum_{i=1}^{n} \pi_i \leq c^* = V(PVC) \qquad (2.36)$$

En dénotant le membre gauche de (2.36) par $W(\pi)$, on obtient que $W(\pi)$ est une borne inférieure valide sur V(PVC) pour chaque vecteur π. La meilleure de ces bornes peut être calculée par:

$$W(\pi^*) = \max_{\pi} \{W(\pi)\} \qquad (2.37)$$

Held et Karp ont donné un exemple où $W(\pi^*)$ est strictement inférieur à V(PVC). Les bornes obtenues par le calcul de $V(PT_\pi)$ sont donc utilisées conjointement avec une méthode de "Branch-and-Bound" pour résoudre le PVC. La description détaillée de l'algorithme de Held et Karp n'est pas nécessaire dans le contexte de notre travail. Soulignons toutefois que par une technique de sous-gradients [62] élaborée par Held et Karp, on peut approximer π^* de façon telle que la borne inférieure obtenue sur V(PVC) lui soit seulement 1% inférieur en moyenne.

Deux ans après la parution de l'algorithme de Held et Karp, Helbig Hansen et Krarup [58] publièrent un article où ils décrivaient une nouvelle version de la méthode de Held et Karp 25 fois plus rapide que cette dernière en moyenne. A l'aide de la version de Helbig Hansen et Krarup de la méthode de Held et Karp, des problèmes de taille n = 80 ont été résolus en un temps moyen de 28.5 secondes (IBM 360-75). Récemment Volgenant et Jonker [123] ont repris l'idée de Held et Karp en apportant des améliorations à la procédure de branchement et au calcul des bornes inférieures par les 1-arbres. Leur algorithme a résolu des PVC symétriques de 100 villes en un temps moyen de 53 secondes (CDC Cyber 750).

2.6 Résolution des PVC symétriques par l'utilisation de la programmation linéaire en nombres entiers

2.6.1 Introduction

Des méthodes similaires à celles présentées dans cette section

pourraient être adaptées au PVC asymétrique. Toutefois, il est peu probable à notre avis que les résultats obtenus soient compétitifs avec ceux des méthodes décrites en 2.3 où la relaxation utilisée est le problème d'affectation. D'ailleurs, à notre connaissance, aucun algorithme résolvant le PVC asymétrique à l'aide de la programmation linéaire en nombres entiers n'est apparu dans la littérature. On supposera donc dans cette section que C est symétrique.

2.6.2 Résolution du modèle de Dantzig et al. [28]

Le modèle de Dantzig et al. [28] du PVC symétrique a été décrit dans la section 2.4. Il contient 3 types de contraintes:

1. des contraintes de degré sur les sommets;
2. des contraintes servant à empêcher l'apparition de sous-tours;
3. des contraintes d'intégrité sur les variables x_{ij} $(i < j)$.

Ces contraintes sont en trop grand nombre pour qu'on puisse résoudre le modèle obtenu dans sa totalité. Pour contourner cette difficulté plusieurs auteurs ont procédé de la façon suivante. Une relaxation linéaire continue du modèle de Dantzig et al. contenant les contraintes de degré et un certain sous-ensemble des contraintes sur les sous-tours est initialement résolue par la méthode du simplex. Si la solution obtenue satisfait aux contraintes relaxées, cette solution est optimale pour le PVC. Dans le cas contraire, cette solution n'est pas entière et/ou le graphe qui lui correspond n'est pas connexe. On ajoute alors au modèle des coupes de Gomory pour imposer l'intégrité sur les variables et/ou des contraintes de (2.24) pour éliminer quelques sous-tours. La solution en main n'étant plus réalisable, des itérations duales du simplex doivent être effectuées pour trouver une autre solution optimale au nouveau modèle linéaire considéré. On procède comme précédemment avec cette dernière en vérifiant l'intégrité des variables et la connexité du graphe et, si nécessaire, en ajoutant d'autres contraintes au modèle.

Le modèle initial se voit donc adjoindre à chaque étape de nouvelles coupes de Gomory ou de nouvelles contraintes de (2.24). Il devient donc de plus en plus près du modèle de Dantzig et al. et la première solution réalisable pour le PVC obtenue est aussi sa solution optimale. Il s'agit donc essentiellement d'une méthode duale de résolution du PVC. L'efficacité d'une telle procédure dépend évidemment grandement du nombre de contraintes qui seront ajoutées au modèle initial.

Il est possible d'imposer l'intégrité sur les variables dans le cadre de la procédure précédente en utilisant une méthode de "Branch-and-Bound". Miliotis [91] ainsi que Christofides et Whitlock [23] ont testé cette alternative. Les résultats qu'ils ont obtenus pour le PVC symétrique sont toutefois inférieurs à ceux fournis par des coupes de Gomory (voir [90]) et nous ne décrirons pas leurs travaux ici. On verra de façon détaillée dans les chapitres 4 et 6 comment utiliser une méthode de "Branch-and-Bound" pour résoudre deux extensions du PVC selon la procédure précédente.

Plusieurs auteurs ont mis au point des méthodes résolvant le PVC de la façon décrite précédemment. Nous résumons ici les travaux des auteurs suivants:

1. Dantzig et al. [28], 1954;
2. Martin [89], 1966;
3. Miliotis [90], 1978;
4. Land [69], 1979;
5. Grötshel [54], 1980;
6. Padberg et Hong [98], 1980;
7. Crowder et Padberg [27], 1980.

Travaux de Dantzig et al. [28]

Dantzig, Fulkerson et Johnson [28] furent les premiers, à notre connaissance, en 1954 à utiliser une procédure similaire à la précédente pour résoudre un PVC de 42 villes. On note toutefois quelques différences.

Tout d'abord leur méthode débute avec une solution de base associée à un tour pour les contraintes de degré et non pas avec la solution optimale du problème contenant ces dernières. Des itérations du simplex sont par la suite effectuées de la façon habituelle afin de diminuer le coût de la solution de base initiale. Toutefois, puisque les coupes de Gomory ne sont pas encore découvertes, une variable hors-base doit être entrée dans la base à un niveau entier. Un sous-tour peut alors survenir. Dantzig et al. ajoute dans ce cas explicitement une contrainte de (2.24) rendant irréalisable la nouvelle solution de base obtenue. Des itérations duales sont par la suite effectuées de façon à obtenir une nouvelle solution de base réalisable entière et correspondant à un tour. Le processus continue de façon analogue avec ce dernier. Après quelques itérations de ce processus, Dantzig et al. ont pu prouver l'optimalité de leur tour par des arguments graphiques et heuristiques.

Travaux de Martin [89]

L'article de Gomory [53] en 1963 marque une autre étape dans l'utilisation de la programmation linéaire en nombres entiers pour résoudre le PVC. Les coupes qu'il a mises au point permettent d'éliminer les solutions fractionnaires optimales que l'on obtient en résolvant par la méthode du simplex un modèle tout entier. Ces coupes conduisirent à de nouveaux algorithmes pour résoudre des problèmes en nombres entiers dont l'algorithme euclidien accéléré de Martin [88]. Ce dernier utilisa son algorithme sur le même PVC que celui traité par Dantzig et al. et d'une manière fort similaire à ceux-ci: il trouve en premier lieu la solution optimale entière pour un ensemble de contraintes initiales composées des 42 contraintes de degré et de 41 contraintes sur les sous-tours, celles-ci ayant été choisies de façon à éliminer les sous-tours les plus susceptibles de se former. S'il y a encore des sous-tours dans la solution, il engendre les contraintes appropriées pour les éliminer et il trouve ensuite une solution entière satisfaisant aussi ces nouvelles contraintes, ajoutant au besoin des coupes de Gomory. Martin parvint de cette façon à trouver le tour optimal pour le problème

de Dantzig et al. en 353 itérations du simplex, engendrant 10 coupes de Gomory et 9 contraintes d'élimination de sous-tours en plus de ses 83 contraintes initiales. Si on se réfère au nombre possible de contraintes dans un PVC de 42 villes, c'est un véritable succès.

Travaux de Miliotis [90]

En 1975, Miliotis [90] tente de nouveau une approche semblable à celle de Dantzig et al. pour le PVC. Il utilise une version en arithmétique entière qu'il a mise au point pour le programme MIF contenu dans le code de Land et Powell [71]. MIF a été décrit en 1.6.

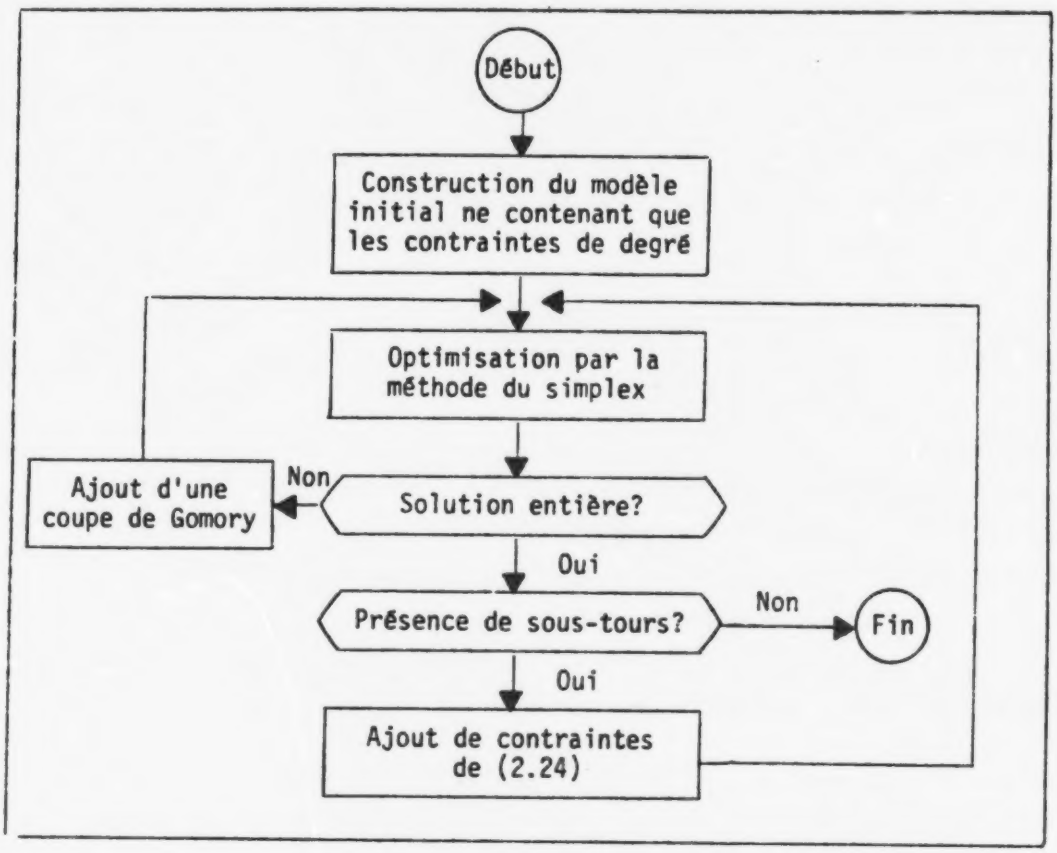
Miliotis implantera et testera 2 variantes de la procédure précédente pour résoudre le PVC. Dans chacune de celles-ci, il relaxe au début les contraintes d'intégrité sur les variables ainsi que les contraintes servant à éliminer les sous-tours ne conservant que les contraintes de degré. Ces deux variantes se différencient par l'ordre dans lequel l'intégrité sur les variables et la connexité du graphe sont obtenues. Dans la première variante l'intégrité est tout d'abord obtenue et ensuite la connexité du graphe. La deuxième variante est tout simplement l'inverse de la première. L'organigramme de la figure 2.1 illustre la première variante de Miliotis.

Les 2 variantes de Miliotis sont très similaires à la méthode de Martin [89]. Contrairement à ce dernier toutefois, l'utilisation du code de Land et Powell permet d'effectuer tout le travail dans un même passage sur ordinateur. La deuxième variante s'est avérée supérieure à la première. Miliotis a pu résoudre avec celle-ci un PVC de 90 villes en 34 secondes sur une CDC 7600.

Miliotis n'engendre qu'une seule coupe pour éliminer une solution fractionnaire. En fait, pour chaque variable prenant une valeur fractionnaire dans la solution considérée, on peut calculer une coupe de Gomory rendant irréalisable cette dernière. Miliotis devait donc choisir parmi les coupes possibles celle qui est la "meilleure".

FIGURE 2.1

Algorithmes de Miliotis: première variante



Miliotis se base sur un certain critère de profondeur que l'on peut décrire ainsi. Dans le code de Land et Powell les coupes sont calculées en termes des variables originales. Soit x le vecteur des variables x_{ij} et $\bar{x}$ la valeur de x pour la solution analysée. On peut définir une coupe de Gomory comme une fonction linéaire f(x) telle que:

1. ses coefficients sont entiers;
2. f(x) est maximisée en $\bar{x}$ (dans [71] il s'agit d'une maximisation);
3. la valeur que prend f(x) évaluée en $\bar{x}$ n'est pas entière.

Soit $f(\bar{x}) = d_o + f_o$ avec d_o un entier et $0 < f_o < 1$.

La contrainte suivante est valide et constitue une coupe de Gomory pour la solution en main:

$$f(x) \leq d_o \tag{2.38}$$

Notons que $\bar{x}$ ne satisfait pas à (2.38).

Il n'est pas facile de trouver une fonction f(x) possédant les trois propriétés précédentes. Le lecteur est référé à [71] pour voir comment on peut calculer une telle fonction à partir de toute rangée du tableau du simplex correspondant à une variable de base prenant une valeur fractionnaire.

Pour quantifier la profondeur d'une coupe Miliotis utilise le rapport suivant:

$$\lambda = \frac{f_o}{d_o} \tag{2.39}$$

La quantité λ évalue la portion du polytope associé au problème linéaire considéré qui se trouve "coupée" par (2.38). Plus λ sera grande, plus "forte" sera la coupe.

Mentionnons que procéder en arithmétique entière permet à Miliotis de connaître la valeur du déterminant de la matrice inverse de la matrice de base et d'engendrer plusieurs coupes à partir de la même variable fractionnaire, la structure de groupe des coupes de Gomory rendant ceci possible (voir [90,92]).

Miliotis a construit la sous-routine INTCON pour effectuer les calculs sous-jacents à la génération d'une coupe de Gomory. Nous utiliserons avec modifications mineures cette sous-routine dans les algorithmes des chapitres 3 et 5.

Travaux de Land [69]

Land [69] en 1979 a amélioré la deuxième variante de Miliotis en

lui incorporant les modifications suivantes. Tout d'abord le problème linéaire contenant uniquement les contraintes de degré est résolu sur un sous-ensemble des variables x_{ij}. Celles qui sont choisies prennent la valeur 1 dans un tour trouvé de façon heuristique ou bien sont susceptibles de faire partie du tour optimal. Par la suite la procédure de Miliotis est exécutée sur ce sous-ensemble de variables tant que des contraintes sont ajoutées au modèle pour imposer l'intégrité des variables ou la connexité du graphe. Lorsqu'un tour optimal est trouvé, les coûts relatifs des variables non considérées sont calculés et s'il n'y a pas optimalité globale, d'autres variables sont ajoutées au modèle et la procédure continue de façon similaire. Notons que Land utilise quelquefois des contraintes de "2-matching", en plus des coupes de Gomory, pour éliminer une solution fractionnaire (voir [69]).

L'algorithme de Land s'inspire surtout des travaux de Dantzig et al. [28] et de ceux de Miliotis [90]. Plusieurs problèmes de 100 villes ont été résolus optimalement par cet algorithme.

Travaux de Grötshel [54], Padberg et Hong [98], Crowder et Padberg [27]

Grötshel et Padberg [55] ont analysé la structure faciale de l'enveloppe convexe des tours du PVC symétrique. Soit $m = \frac{1}{2}n(n-1)$ et $x \in R^m$ un vecteur des variables x_{ij} où $1 \leq i<j \leq n$. Soit Q_t l'enveloppe convexe des $x \in R^m$ tels que le graphe associé à x est un tour. Il n'existe présentement aucune caractérisation linéaire avec variables continues pour Q_t. Toutefois certains résultats intéressants sont connus au sujet de Q_t. Grötshel et Padberg ont démontré en particulier que chacune des contraintes de (2.24) définit une facette[1] de Q_t si $n \geq 4$. Pour y arriver ils ont démontré qu'il y a exactement $q = \frac{1}{2}(n^2-3n)$ tours différents qui satisfont à égalité une contrainte de (2.24) donnée. Or la dimension de Q_t est précisément q si $n \geq 3$ [96].

(1) La définition de termes tels enveloppe convexe, facette, etc. est donnée dans [115].

Ce résultat explique pourquoi fréquemment très peu de coupes de Gomory sont nécessaires pour atteindre une solution entière lorsqu'on utilise une procédure comme celle décrite au début de 2.6.2 pour résoudre le PVC symétrique. En fait, chaque contrainte de (2.24) ajoutée au modèle, en plus de forcer la connexité sur le graphe, sert aussi à imposer l'intégrité sur les variables car elle constitue une facette de Q_t.

Il existe, à part (2.24), d'autres contraintes pour éliminer les sous-tours. Chvatal [24] a mis au point une nouvelle forme de telles contraintes appelées les contraintes de peigne(1). Grötshel et Padberg [55] ont par la suite généralisé les contraintes de Chvatal pour en arriver aux contraintes de peigne généralisées. L'aspect intéressant de ces dernières est qu'elles constituent elles aussi, tout comme les contraintes de Chvatal, des facettes de Q_t, tel que l'ont démontré Grötshel et Padberg [55]. La description détaillée des contraintes de peigne ou des contraintes de peigne généralisées ne sera pas effectuée ici car elles n'ont pas été utilisées dans nos travaux.

A la suite des travaux de Grötshel et Padberg sur Q_t, l'utilisation des contraintes de peigne généralisées (incluant les contraintes de Chvatal) devint aussi possible dans la procédure de résolution du PVC décrit au début de 2.6.2. Ainsi, en 1980, Grötshel [54] les utilise pour résoudre un PVC de 120 villes. Par la suite Padberg et Hong [98] ont donné des résultats de tests sur 74 problèmes contenant 15 à 318 villes. L'optimalité a pu être prouvée pour 58 d'entre eux. En outre, pour le problème de 318 villes, le seul de taille supérieure à 120 parmi ceux testés, un tour à 1% optimal a été trouvé. Finalement Crowder et Padberg [27] ont repris l'analyse des résultats de [98] en effectuant aussi de nouveaux passages sur ordinateur pour le problème de 318 villes. Les itérations ont débuté avec le tour à 1% optimal précédemment trouvé et l'optimalité a finalement été atteinte

(1) "Comb inequalities".

Il s'agit à notre connaissance du PVC symétrique de plus grande taille jamais résolu optimalement.

2.7 Conclusion

Notre résumé des connaissances actuelles au sujet du PVC a surtout porté sur les principaux modèles qui ont été mis au point pour le formuler mathématiquement et sur les algorithmes optimaux les plus performants pour le résoudre. Nous voulons souligner trois constatations importantes:

1. le modèle de Dantzig et al. [28] a fortement inspiré les chercheurs dans l'élaboration d'algorithmes pour le PVC;
2. la relaxation qui semble la meilleure pour le PVC asymétrique est celle du problème d'affectation et les algorithmes basés sur cette relaxation [2,15] sont les plus efficaces pour le PVC asymétrique;
3. dans le cas du PVC symétrique, la relaxation provenant des 1-arbres [60,61,123] et la programmation linéaire en nombres entiers sont les deux grandes approches les plus efficaces pour résoudre ce problème.

Le principal avantage des méthodes décrites en 2.6 est leur flexibilité. Même si dans un modèle pour une extension du PVC les contraintes de degré et les contraintes d'élimination de sous-tours illégaux sont différentes des contraintes correspondantes dans un modèle du PVC, on peut encore facilement utiliser un algorithme similaire à ceux décrits en 2.6 pour résoudre un tel modèle. De plus la grande efficacité de l'approche basée sur la programmation linéaire en nombres entiers pour résoudre le PVC symétrique nous permet d'espérer de bons résultats sur des extensions du PVC avec une telle approche. Nos chapitres 3 et 5 où des coupes de Gomory sont utilisées pour maintenir l'intégrité et nos chapitres 4 et 6 où une méthode de "Branch-and-Bound" sert à cette fin viennent tous confirmer la justesse de cette hypothèse.

CHAPITRE 3

LE PROBLEME DES m VOYAGEURS DE COMMERCE (m-PVC)

3. LE PROBLEME DES m VOYAGEURS DE COMMERCE (m-PVC)

3.1 Introduction

Une des généralisations du PVC les plus étudiées est le problème des m voyageurs de commerce (m-PVC) [1,45,47,119] dont l'appellation anglaise la plus usuelle est "the multiple travelling salesmen problem"(1). Le m-PVC est habituellement défini de la façon suivante: il faut trouver les routes de m voyageurs de commerce qui, partant tous d'une même ville appelée leur dépôt. doivent visiter (n-1) autres villes puis revenir à leur point de départ. Chaque ville n'est visitée qu'une seule fois et par un seul commis sauf le dépôt qui est visité m fois(2). Dans ce chapitre nous identifierons le dépôt à la ville 1(3) et les villes à visiter seront dénotées par 2,...,n-1. N = {1,...,n} dénotera l'ensemble des villes d'un m-PVC.

Il existe plusieurs types de m-PVC. Ils se différencient les uns des autres par la liberté accordée à m et conséquemment par le critère de performance à optimiser lors de la construction des m routes. Pour certains [1,47,119] le nombre de commis à utiliser est fixé a priori à une certaine constante $\bar{m}$ ($1 \leq \bar{m} \leq n-1$). Dans ce cas nous utiliserons explicitement la notation $\bar{m}$-PVC. Pour ce problème le critère de performance est la somme des coûts de transport c_{ij} où l'arc (ou arête) (i,j) est suivi par un des commis. On suppose ici que c_{ij} est indépendant du commis particulier effectuant le trajet de la ville i à la ville j(4). En identifiant chaque commis-voyageur à un véhicule donné, cette hypothèse est équivalente à supposer que les véhicules sont identiques.

Quelques autres auteurs définissent le m-PVC en laissant m libre [6,64]. Ceci peut se justifier par le fait que le coût total

---

(1) Aussi "the multisalesmen problem".

(2) On commet volontairement ici un abus de langage car la ville-dépôt n'est pas visitée au même sens qu'une ville où un certain service est rendu par le commis.

(3) Sauf dans la section 3.2.

(4) Il faut alors utiliser des variables x_{ij}^k et le problème est plus difficile à résoudre.

des m routes est influencé, en plus de leurs villes constituantes, par leur nombre. De plus, dans certaines applications du m-PVC, le nombre optimal de commis à utiliser doit aussi être déterminé. Ainsi, dans certains cas, m est libre mais borné supérieurement; nous utiliserons alors $\bar{m}$ comme une borne supérieure sur m. Notons qu'une telle borne supérieure existe toujours et est à la limite égale à n-1.

Donner à m la possibilité de varier permet d'introduire dans l'objectif, en plus du critère habituel du coût total du transport, un coût fixe pour chaque commis utilisé[1]. Nous supposerons dans ce chapitre que ce coût est identique pour chaque commis[2] et nous le dénoterons par g.

Afin d'éviter toute ambiguïté de notation,

1. la notation $\bar{m}$-PVC signifie que m est fixé à $\bar{m}$;
2. la notation m_ℓ-PVC signifie que $1 \leq m \leq \bar{m}$[3];
3. la notation m-PVC désigne le problème général où m est soit libre, soit fixé à $\bar{m}$.

Le terme m-tour servira à dénommer toute solution réalisable d'un m-PVC. En termes de la théorie des graphes (section 1.4) un m-tour est constitué par m circuits (cycles) tels que:

1. chaque sommet autre que le dépôt du graphe G = (N,A) associé au m-PVC fait partie d'exactement un de ces m circuits (cycles);
2. le sommet-dépôt est contenu dans chacun de ces m circuits (cycles).

---

(1) Hong et Padberg [64] utilisent dans ce cas l'appellation "le problème des m voyageurs de commerce avec coûts fixes ("fixed charges").

(2) Certains auteurs utilisent des coûts possiblement différents sur les commis. Ceci n'introduit pas de difficultés supplémentaires comme nous le verrons.

(3) Il est évidemment possible de définir le m_ℓ-PVC avec $m \leq m \leq \bar{m}$ où $1 \leq m \leq \bar{m}$. Ceci n'introduit pas de difficultés supplémentaires dans notre algorithme.

Le m-PVC constitue donc la recherche d'un m-tour de poids minimal.

Le PVC est un cas particulier du m_ℓ-PVC et dans ce sens les applications du m-PVC sont à tout le moins aussi nombreuses que celles du PVC. Lenstra et Rinnoy Kan [83] mentionnent les deux suivantes[1]:

1. construction de routes pour une équipe d'hommes ("commis") devant visiter des boîtes téléphoniques ("villes") dans le nord de la Hollande;
2. calcul du nombre minimum de véhicules ("commis") pouvant visiter 200 boîtes postales ("villes") dans la ville d'Utrecht.

Le m-PVC est à son tour un cas particulier de plusieurs autres problèmes, en particulier du c.m-PVC considéré au chapitre 4. Son étude, en plus de conduire à des applications immédiates, revêt donc une grande importance pour le traitement de d'autres extensions du PVC.

Les liens théoriques entre le PVC et le m-PVC sont nombreux. Ainsi lorsque le graphe associé au m-PVC est complet, ce problème se réduit au PVC dans le cas où m est libre et dans le cas où la matrice des coûts de transport C est symétrique et euclidienne au dépôt, c'est-à-dire:

$$c_{1j} \leq c_{1i} + c_{ij} \qquad (i,j = 2,\ldots,n) \quad (3.1)$$

De plus on verra dans la prochaine section que par de simples modifications de la matrice des coûts de transport on peut transformer un m-PVC en un PVC et ce que C soit symétrique ou asymétrique, que m soit libre ou fixé et même que les coûts fixes des commis soient identiques ou non. C'est probablement d'ailleurs pour cette raison que, pendant très longtemps, les chercheurs se sont concentrés sur le PVC, négligeant le développement d'algorithmes spécifiques au m-PVC. Toutefois, notre algorithme pour le m-PVC

(1) En fait Lenstra et Rinnoy Kan résolvent ces deux problèmes à l'aide d'un heuristique pour le PVC. Pour y arriver, ils les transforment tout d'abord en un PVC par leur transformation que nous verrons en 3.2.

et celui de Gavish et Srikanth [47] démontrent clairement que d'un point de vue algorithmique, à tout le moins dans le cas symétrique, il faut traiter le m-PVC séparément du PVC. On verra d'ailleurs ce qui justifie cette affirmation dans les prochaines sections.

Même si le m-PVC peut se transformer de façon artificielle en un PVC, il existe des différences fondamentales entre ces deux problèmes:

1. résoudre le m-PVC nécessite essentiellement la prise de deux décisions distinctes; d'une part il faut décider quelles seront les villes sur une même route et d'autre part, pour une route donnée dont les villes constituantes sont déterminées, il faut trouver dans quel ordre ces villes seront visitées[1]. Ces deux[2] décisions doivent être prises simultanément et non de façon séquentielle pour assurer l'optimalité de la solution obtenue;
2. le choix du dépôt influence la solution optimale obtenue, ce qui n'est pas le cas pour le PVC. Dans le m-PVC on suppose connue la "localisation" du dépôt. Cette hypothèse sera relaxée dans le chapitre 6 de ce texte.

Le plan de notre troisième chapitre est le suivant:

- section 3.2: revue de quelques transformations du m-PVC en un PVC;
- section 3.3: revue des modèles asymétriques du m-PVC;
- section 3.4: revue des modèles symétriques du m-PVC;
- section 3.5: revue des algorithmes optimaux pour le m-PVC;
- section 3.6: présentation de deux nouveaux algorithmes pour le m-PVC;
- section 3.7: comparaison de nos algorithmes pour le m-PVC par rapport à ceux déjà existants;
- section 3.8: conclusion.

---

(1) Ce qui constitue la résolution d'un PVC.

(2) Pour le m_ℓ-PVC il faut aussi décider du nombre de routes à construire.

3.2 Transformations du m-PVC en un PVC

3.2.1 Introduction

On peut toujours transformer un m-PVC en un PVC. Bien que nos recherches ne concernent pas la construction de telles transformations[1], il nous semble intéressant d'introduire dans ce texte une revue de la littérature à leur sujet. Tout d'abord certains algorithmes pour le m-PVC utilisent de telles transformations. De plus nous n'avons rencontré nul part un tel résumé dans la littérature et finalement nous utilisons une de ces transformations pour la construction d'une borne supérieure au problème traité dans le chapitre 6 de cette étude.

La notation utilisée dans cette section a été introduite en 3.1 ainsi qu'au chapitre 1. Notons toutefois que pour faciliter l'écriture des transformations, le dépôt sera situé exceptionnellement à la ville n dans cette section.

Nous avons relevé dans la littérature 5 transformations différentes du m-PVC en un PVC. Trois paramètres permettent de les distinguer:

1. elles concernent un m-PVC asymétrique ou bien un m-PVC symétrique;
2. m est soit libre, soit fixé;
3. les coûts fixes des commis sont identiques ou non.

Quelques commentaires s'imposent ici à propos de ces 3 paramètres. Puisqu'à la rigueur tout m-PVC symétrique peut être considéré comme un m-PVC asymétrique, toute transformation concernant un m-PVC asymétrique peut être utilisée dans le cas symétrique. Toutefois, ce n'est pas nécessairement souhaitable car on a vu dans le chapitre sur le PVC que,

---

(1) Dans cette section "transformation" signifie "transformation d'un m-PVC en un PVC".

d'un point de vue algorithmique, le PVC symétrique doit être traité différemment du PVC asymétrique. Dans la suite du texte, lorsque nous dirons qu'une transformation traite du m-PVC asymétrique, il sera toujours sous-entendu qu'à la rigueur elle peut aussi être employée pour le cas symétrique.

Lorsque m est fixé, il est inutile de considérer dans l'objectif des coûts fixes pour les commis et toute transformation où m est fixé est nécessairement plus limitée qu'une transformation où m peut varier. Il est cependant possible d'utiliser une transformation pour m fixé dans le cas où m est libre en faisant varier successivement m de 1 jusqu'à $\bar{m}$. Procéder ainsi pour résoudre un m_ℓ-PVC conduirait à la résolution de $\bar{m}$ problèmes du type PVC et n'est évidemment pas à conseiller vu l'existence de transformations traitant directement du cas du m_ℓ-PVC. Encore ici lorsqu'on mentionnera qu'une transformation présuppose que m est fixé, il faudra se rappeler qu'à la limite on peut aussi l'utiliser si m est libre. Toutefois, pour toute transformation de ce type il n'y aura pas de coûts fixes.

Ces 5 transformations sont plus ou moins faciles à comprendre selon le type de m-PVC auquel elles correspondent. La transformation la plus simple concerne le $\bar{m}$-PVC asymétrique et, à notre avis, la plus difficile à visualiser traite du m_ℓ-PVC avec coûts fixes non identiques. Elles ont plusieurs liens communs entre elles et il nous semble important de bien les faire comprendre au lecteur avant de décrire ces transformations de façon plus technique.

L'idée de base de toutes ces transformations est la suivante: dans un m-PVC toutes les villes sont visitées une fois exactement, sauf la ville-dépôt qui l'est m fois. En ajoutant artificiellement au problème un certain nombre de copies du dépôt qui devront être visitées chacune une fois exactement, on "simule" m visites au dépôt par une visite à m dépôts différents. Ainsi dans le problème augmenté de villes artificielles, chaque ville est visitée une seule fois, ce qui est le cas

pour un PVC. La première transformation expliquée dans la section 3.2.2 est identique à cette idée de base et nous donnerons alors une illustration plus détaillée de son application. Toutes les transformations que nous verrons impliquent donc toutes une augmentation du nombre de villes à considérer. Elles permettent de résoudre un m-PVC avec matrice de coûts de transport C_{nxn} et coûts fixes sur les commis égaux ou non en résolvant un PVC avec matrice de coûts de transport $C^*_{pxp} = (c^*_{ij})$ où $p > n$, les coûts fixes étant ajoutés sur les arcs ou arêtes adjacents aux villes artificielles. Toute transformation permet ainsi d'écrire un m-tour pour un m-PVC sur n villes comme un tour[1] pour un PVC sur p villes. En termes de la théorie des graphes une transformation fait correspondre un m-tour sur un graphe orienté (non-orienté) G = (N,A) à un tour sur graphe G* = (N*,A*) où:

$$N^* = N \cup N'$$
$$A^* = A \cup A'$$

avec N' = ensemble des sommets associés aux dépôts artificiels adjoints au problème;

A' = ensemble des arcs (arêtes) incidents à un sommet de N'.

Donc, C* est la matrice des distances associée au PVC sur G*. La matrice C* sera construite à partir de C et des coûts fixes sur les commis.

La preuve de la validité d'une transformation donnée doit donc démontrer qu'un m-tour est optimal pour le m-PVC considéré si et seulement si le tour qu'on lui fait correspondre par la transformation est optimal pour le PVC équivalent.

Toute transformation sera complètement caractérisée en spécifiant:

1. le (les) cas particulier(s) du m-PVC qu'elle traite;

---

(1) Toute solution réalisable d'un PVC (voir le chapitre 2).

2. la matrice, dite de transformation, C* qui lui est associée.

Une idée de la preuve de validité de la transformation sera donnée pour certaines transformations. Nous comparerons de façon brève leur applicabilité respective en termes du nombre de cas particuliers du m-PVC qu'elles peuvent traiter.

3.2.2 La transformation d'Eilon et al. [35]

La première transformation à apparaître dans la littérature parmi celles que nous avons relevées est celle d'Eilon et al. [35] datant de 1971. Elle traite du $\bar{m}$-PVC asymétrique et correspond directement à l'idée de base de ces transformations que nous avons expliquée en 3.2.1. On adjoint au problème $\bar{m}$-1 dépôts artificiels qui correspondent aux villes n+1 jusqu'à n+$\bar{m}$-1. La matrice C^*_{pxp} avec p = n+$\bar{m}$-1 qui est associée à la transformation d'Eilon et al. est donnée par les équations suivantes[1]:

$$
(3.2)\quad
\begin{cases}
c^*_{ij} = c_{ij} & (i,j = 1,\ldots,n) \qquad (3.2\text{-}1)\\
c^*_{ij} = \infty & (i = n,\ldots,p;\ j = n+1,\ldots,p) \qquad (3.2\text{-}2)\\
c^*_{ij} = \begin{cases} c_{nj} & (i = n+1,\ldots,p;\ j = 1,\ldots,n-1) \qquad (3.2\text{-}3)\\ c_{in} & (i = 1,\ldots,n-1;\ j = n+1,\ldots,p) \qquad (3.2\text{-}4)\end{cases}\\
c^*_{ij} = \infty & (i = n+1,\ldots,p;\ j = n) \qquad (3.2\text{-}5)
\end{cases}
$$

La construction de C* est basée sur les constatations suivantes. Tout passage d'une ville artificielle n+i (i = 1,...,$\bar{m}$-1) à une des villes initiales j autre que le dépôt (j = 1,...,n-1) s'interprète comme un passage du dépôt à la ville j; le coût de ce trajet est donc c_{nj} (éq. 3.2-3). On peut donner une explication similaire au coût du passage de la ville i (i = 1,...,n-1) à la ville artificielle n+j (j = 1,...,$\bar{m}$-1) (éq. 3.2-4). Les coûts c^*_{ij} auxquels on donne une valeur ∞ sont associés à un passage d'un dépôt à un autre, qu'il soit artificiel ou non. Un tel trajet est prohibé car il s'interprète comme un passage du dépôt à lui-

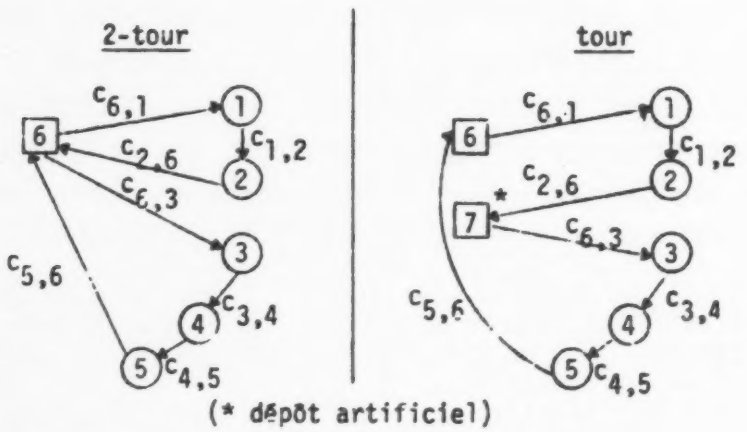
---

(1) Dans toutes les égalités données dans cette section, il sera toujours sous-entendu que $c^*_{ii} = \infty$ pour tout i.

même (éq. 3.2-2 et 3.2-5). La figure 3.1 illustre un 2-tour pour un 2-PVC avec n = 6 et le tour qui lui est associé par la transformation d'Eilon et al. Les coûts de transport sur le 2-tour et le tour sont indiqués sur les arcs.

FIGURE 3.1

Correspondance entre un 2-tour et un tour par la transformation d'Eilon et al.



(* dépôt artificiel)

3.2.3 La transformation de Lenstra et Rinnoy Kan [83]

La seconde transformation que nous voulons expliciter a été établie en 1975 par Lenstra et Rinnoy Kan [83]. Leur transformation généralise celle d'Eilon et al. et peut traiter, pour une matrice de coûts asymétriques, un m_ℓ-PVC avec coûts fixes tous égaux et aussi un $\bar{m}$-PVC. Les équations (3.3) illustrent la construction de C* pour la transformation de Lenstra et Rinnoy Kan avec comme précédemment $p = n+\bar{m}-1$:

$$
(3.3)\quad
\begin{cases}
c^*_{ij} = c_{ij} & (i,j = 1,\ldots,n) \qquad (3.3\text{-}1)\\
c^*_{ij} = \lambda & (i = n,\ldots,p;\ j = n+1,\ldots,p) \qquad (3.3\text{-}2)\\
c^*_{ij} = \lambda & (i = n+1,\ldots,p;\ j = n) \qquad (3.3\text{-}3)\\
c^*_{ij} = \begin{cases} c_{nj} & (i = n+1,\ldots,p;\ j = 1,\ldots,n-1) \qquad (3.3\text{-}4)\\ c_{in} & (i = 1,\ldots,n-1;\ j = n+1,\ldots,p) \qquad (3.3\text{-}5) \end{cases}
\end{cases}
$$

Cette transformation s'explique de façon identique à celle d'Eilon et al. sauf pour les passages d'un dépôt à un autre (distinct) qui, ici, sont permis en autant que leurs coûts, dénotés par λ, soient différents de $-\infty$ (éq. 3.3-2 et éq. 3.3-3). Pour donner une interprétation de λ on peut raisonner de la façon suivante. On doit résoudre un PVC asymétrique sur la matrice C* donnée par (3.3). Les sommets associés à ce PVC, numérotés $1,2,\ldots,n+\bar{m}-1$, auront donc tous un degré intrant de 1 et un degré extrant de 1, en particulier les $\bar{m}$ sommets associés aux dépôts dénotés $n,\ldots,n+\bar{m}-1$. Puisque dans un m-PVC le tour d'un commis doit débuter et se terminer au dépôt, dans le PVC associé le tour d'un commis débute et se termine à un des $\bar{m}$ dépôts artificiels[1]. Clairement il y aura donc au maximum $\bar{m}$ tours représentés dans le tour d'un PVC. De plus, chaque fois qu'un lien inter-dépôt est présent dans un tour du PVC, on diminue d'une unité le nombre de tours pour un commis qu'on peut représenter. On en arrive donc à la conclusion que si la solution du PVC sur la matrice C* contient $\bar{m}$-a liens entre les dépôts artificiels alors dans la solution du m-PVC associée on aura $m = a$. Le coût du passage d'un dépôt à un autre, diminuant d'une unité le nombre de commis utilisés, diminue donc de g le coût total dû aux coûts fixes des commis et ce coût doit donc être -g d'où la conclusion que $\lambda = -g$. Résoudre un PVC sur C* est équivalent à résoudre un:

1. $\bar{m}$-PVC <u>si $\lambda = -\infty$</u> car alors $g = -\infty$;

2. m_ℓ-PVC où m est minimisé en premier lieu et le coût des tours en deuxième lieu si $\lambda = -\infty$ car $g = \infty$;

3. m_ℓ-PVC <u>si $\lambda = -g \in (0,\infty)$</u>.

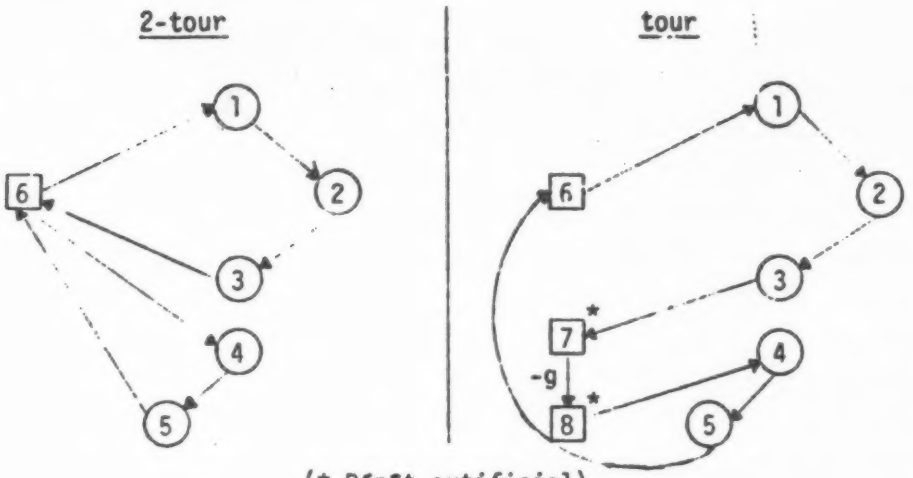
Dans la figure 3.2 on donne l'exemple d'un 2-tour, solution réalisable d'un m_ℓ-PVC où $\bar{m} = 3$, $n = 6$ et du tour qui lui correspond par la transformation de Lenstra et Rinnoy Kan. On remarquera que l'utilisation de l'arc (7,8) diminue de g la somme totale des coûts fixes.

---

(1) Evidemment nous entendons le dépôt plus ses $\bar{m}$-1 copies.

FIGURE 3.2

Illustration de la transformation
de Lenstra et Rinnoy Kan



(* Dépôt artificiel)

Mentionnons en dernier lieu au sujet de la transformation de Lenstra et Rinnoy Kan qu'il est facile de démontrer que leur transformation peut aussi tenir compte d'une borne inférieure sur m, dénotée par $\underline{m}$. Il suffit pour y arriver de poser dans C* le coût de $\underline{m}$ liens inter-dépôts à la valeur $+\infty$. Il est clair qu'une transformation pour un $\bar{m}$-PVC pourrait aussi servir pour résoudre un m_ℓ-PVC où la contrainte supplémentaire $m \geq \underline{m}$ serait présente. Il faudrait alors résoudre $(\bar{m}-\underline{m})+1$ problèmes de type PVC, ce qui serait évidemment beaucoup plus long.

3.2.4 La transformation de Bellmore et Hong [6]

La transformation que nous voulons présenter ici ainsi que les deux suivantes permettent de traiter le cas où les coûts fixes des commis ne sont pas identiques[1]. Nous supposerons lors de leur description que les $\bar{m}$ commis sont indexés par $n,\ldots,n+\bar{m}-1$[2], que leurs coûts

(1) A fortiori elles peuvent traiter du cas où les coûts fixes sont identiques.

(2) Nous avons préféré indexer les commis par $n,\ldots,n+\bar{m}-1$ au lieu de l'indexation plus naturelle $1,\ldots,\bar{m}$ car alors l'indice d'un commis est identique au numéro du dépôt artificiel qui lui correspond.

fixes, dénotés $g_n,\ldots,g_{n+\bar{m}-1}$, sont dans l'ordre non-décroissant, c'est-à-dire:

$$g_n \leq g_{n+1} \leq \ldots \leq g_{n+\bar{m}-1} \tag{3.4}$$

Le coût d'un m-tour sera donc égal à la somme des coûts de transport d'une ville à l'autre impliqués dans les m routes plus la somme des coûts fixes des commis affectés aux m routes. Il est évident que dans tout m-tour optimal on peut toujours assigner les m premiers commis[1] aux m routes construites. Notons que le fait de permettre aux coûts fixes des commis de ne pas être nécessairement identiques peut conduire à un m-tour optimal différent de celui d'un m_ℓ-PVC où les coûts fixes sont tous égaux même si la matrice des coûts de transport est identique pour les deux problèmes.

Au niveau des transformations, le traitement des coûts fixes non identiques est plus difficile que s'ils sont égaux entre eux. La transformation décrite ici, celle de Bellmore et Hong [6], permettra d'illustrer la raison de cette affirmation. Le transformation de Bellmore et Hong donnée en 1974 concerne un m-PVC asymétrique et peut être utilisée que m soit libre ou fixé. Comme les deux transformations précédentes, on ajoute au problème $\bar{m}$-1 dépôts artificiels dénotés par les villes $n+1,\ldots,n+\bar{m}-1=p$. La matrice de transformation de Bellmore et Hong est donnée par les équations (3.5):

$$(3.5)\quad \begin{cases} c^*_{ij} = c_{ij} & (i,j = 1,\ldots,n-1) & (3.5\text{-}1)\\ c^*_{ij} = c_{nj} + \frac{1}{2}g_i & (i = r,\ldots,p;\ j = 1,\ldots,n-1) & (3.5\text{-}2)\\ c^*_{ij} = c_{in} + \frac{1}{2}g_j & (i = 1,\ldots,n-1;\ j = n,\ldots,p) & (3.5\text{-}3)\\ c^*_{i,i-1} = \frac{1}{2}g_{i-1} - \frac{1}{2}g_i & (i = r+1,\ldots,p) & (3.5\text{-}4)\\ c^*_{ij} = \infty & (\text{ailleurs}) & (3.5\text{-}5) \end{cases}$$

L'équation (3.5-1) est évidente. L'équation (3.5-2) indique que le passage d'un des dépôts ($i = n,\ldots,p$) vers une des villes originales ($j = 1,\ldots,n-1$) dans le PVC s'interprète comme le passage du dépôt vers

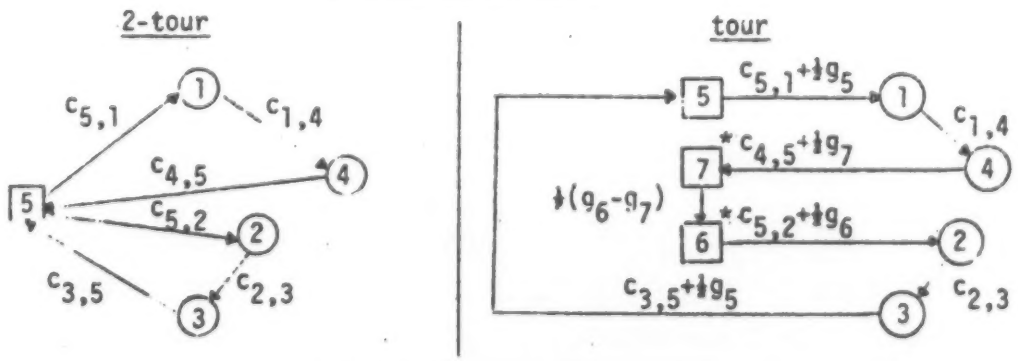
(1) C'est-à-dire ceux indexés par $n, n+1,\ldots,n+m-1$.

une des villes originales dans le m-PVC. Ce passage implique aussi l'utilisation du commis i et la moitié du coût fixe g_i est ajoutée en sus de c_{nj}. L'équation (3.5-5) révèle le fait que, dans le PVC, tout passage d'un dépôt, disons i, à un autre, noté j, est interdit à moins que j = i-1. De plus par (3.5-4) on voit qu'un tel passage d'un dépôt i au dépôt i-1 signifie la non-utilisation du commis i (processus similaire dans la transformation de Lenstra et Rinnoy Kan); la moitié du coût fixe ($\frac{1}{2}g_i$) doit donc être soustraite puisque $\frac{1}{2}g_i$ a déjà été additionné auparavant, une entrée dans le sommet i ayant dû être effectuée plus tôt pour qu'on puisse en sortir. Au sujet de (3.5-4) mentionnons en outre que le coût $\frac{1}{2}g_{i-1}$ sera effectivement présent dans le coût total du tour seulement si en partant du dépôt i-1 on se dirige vers une des villes originales (j=1,...,n-1) sinon cela signifie qu'on se dirige vers i-2 et alors $\frac{1}{2}g_{i-1}$ sera soustrait. L'équation (3.5-3) s'explique de façon similaire à (3.5-2) à la nuance près que lorsqu'on passe de la ville originale i au dépôt j, cela ne signifie pas que la ville i est sur le tour du commis j; $\frac{1}{2}g_j$ est toutefois additionné au coût de ce passage car possiblement (si on ne se dirige pas vers j-1) le commis j sera utilisé.

Pour illustrer ces commentaires au sujet de la transformation de Bellmore et Hong nous donnons dans la figure 3.3 l'exemple d'un 2-tour pour un m_ℓ-PVC avec n = 5, $\bar{m}$ = 3 et l'exemple du tour qui lui correspond par leur transformation.

FIGURE 3.3

Illustration de la transformation de Bellmore et Hong



(*dépôt artificiel)

En supposant que les commis 5 et 6[1] sont utilisés dans le 2-tour, le coût total du 2-tour (coûts de transport + coûts fixes) est égal au coût du tour. On voit d'ailleurs que la somme des coûts fixes dans le tour est g_5+g_6. L'utilisation de l'arc (7,6) a permis d'une part de diminuer d'une unité le nombre de commis utilisés et signifie aussi que c'est le commis 7 qui sera non utilisé.

Afin de démontrer la validité de leur transformation, Bellmore et Hong prouvent dans l'ordre les résultats suivants:

1. pour tout m-tour sur G il existe un tour sur G* de coût égal (la figure 3.3 nous permet de nous en convaincre);
2. pour tout tour sur G*, il existe un m-tour sur G de coût inférieur ou égal;
3. par les résultats 1 et 2 on a donc que tout m-tour correspondant à un tour optimal sur G* est de coût égal à ce tour;
4. par 3, tout m-tour correspondant à un tour optimal sur G* est lui-même optimal pour le m-PVC sur G.

Le résultat 2 indique que les coûts fixes présents dans un tour sur G* ne correspondent pas nécessairement aux coûts fixes les plus petits. En gardant les mêmes routes mais en prenant les commis associés aux plus petits coûts fixes pour les parcourir on obtient un m-tour de coût inférieur ou égal.

Mentionnons finalement au sujet de cette transformation qu'en posant $c_{i,i-1} = \infty$ pour $i = n+1,\ldots,p$ on résout clairement un $\bar{m}$-PVC et qu'en posant $c_{i,i-1} = \infty$ pour $i = n+1,\ldots,n+\underline{m}-1$ on peut résoudre un m_ℓ-PVC avec borne inférieure de $\underline{m}$ sur le nombre de commis à utiliser.

3.2.5 La transformation de Hong et Padberg [64]

Les deux dernières transformations que nous verrons ici concernent le cas symétrique d'un m_ℓ-PVC avec coûts fixes non identiques. Dans

(1) En fait les commis ayant les plus petits coûts fixes.

le cas symétrique où le graphe associé au m-PVC est non-orienté, on ne peut distinguer l'entrée à une ville de sa sortie. Il semble donc qu'on ne peut pas, comme Bellmore et Hong le font, tenir compte dans le PVC de la moitié du coût fixe à l'entrée du sommet-dépôt et de l'autre moitié à la sortie. C'est probablement pour cette raison que Hong et Padberg [64], les auteurs de la transformation que nous allons décrire ici, mentionnent que la transformation de Bellmore et Hong ne se généralise pas au cas symétrique. Il s'avère toutefois que cette affirmation est fausse comme le démontrera Rao plus tard. La transformation de Rao [102] sera la dernière transformation que nous verrons dans cette section.

La transformation de Hong et Padberg transforme un m-PVC sur n villes en un PVC sur $n+\bar{m}+4 = p$ villes. La construction de C* à partir de C est basée sur les équations (3.6):

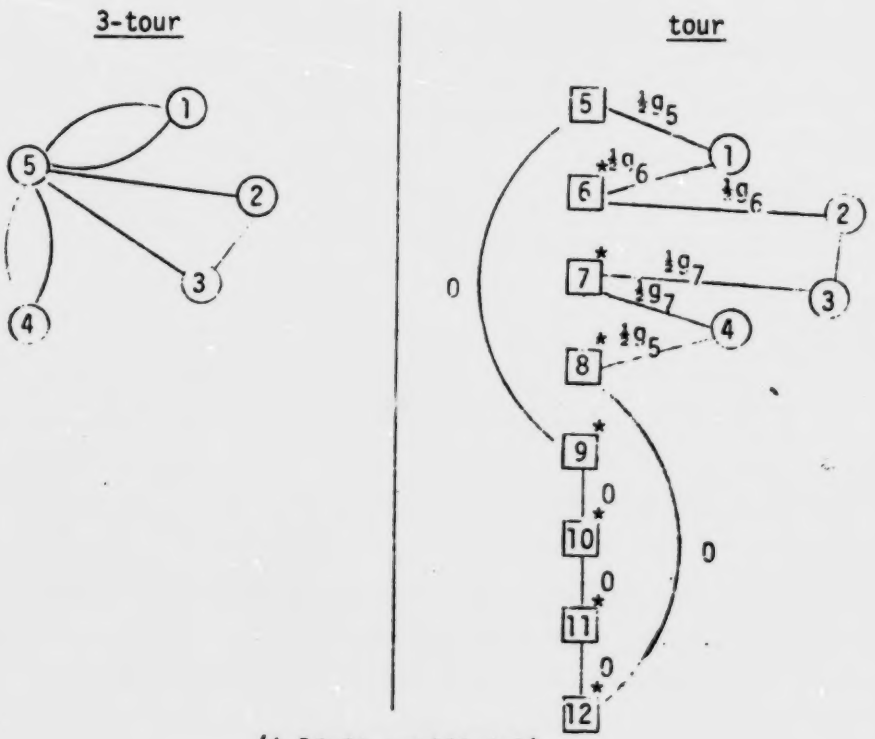
$$
(3.6)\quad
\begin{cases}
c^*_{ij} = c_{ij} & (i,j = 1,\ldots,n-1) & (3.6\text{-}1)\\
c^*_{ij} = c_{in}+\frac{1}{2}g_j & (i = 1,\ldots,n-1;\ j = n,\ldots,n+\bar{m}-1) & (3.6\text{-}2)\\
c^*_{i,n+\bar{m}} = c_{in}+\frac{1}{2}g_n & (i = 1,\ldots,n-1;\ j = n,\ldots,n+\bar{m}-1) & (3.6\text{-}3)\\
c^*_{n,n+\bar{m}+1} = c^*_{n+\bar{m},n+\bar{m}+4} = 0 & & (3.6\text{-}4)\\
c^*_{i,i+1} = 0 & (i = n+1,\ldots,n+\bar{m}-2 \text{ ou } i = n+\bar{m}+1,\ldots,n+\bar{m}+3) & (3.6\text{-}5)\\
c^*_{n+\bar{m}-1,n+\bar{m}+3} = 0 & & (3.6\text{-}6)\\
c^*_{i,n+\bar{m}+2} = 0 & (i = n+1,\ldots,n+\bar{m}-1) & (3.6\text{-}7)\\
c^*_{ij} = \infty & (\text{ailleurs}) & (3.6\text{-}8)
\end{cases}
$$

Comme on peut facilement s'en rendre compte, cette transformation n'est pas aussi "naturelle" que les précédentes et il n'est pas dans notre intention ici de l'expliquer de façon détaillée. Tout lecteur intéressé est référé à l'article de Hong et Padberg. Nous donnerons toutefois les quelques commentaires suivants à son suiet. Considérons un m-tour

donné, solution réalisable d'un m_ℓ-PVC où au maximum $\bar{m}$ commis peuvent être utilisés. Comme on l'a déjà mentionné on doit affecter aux m routes de ce m-tour les commis n,...,n+m-1 pour que la somme des coûts fixes de ce m-tour soit la plus petite possible. Le PVC associé au m_ℓ-PVC considéré contiendra $n+\bar{m}+4$ villes. Le tour associé à ce m-tour dans le PVC est construit de la façon suivante: la ville n est réunie à la ville n+1 par l'intermédiaire de la route du commis n, la ville n+1 est réunie à la ville n+2 par l'intermédiaire de la route du commis n+1 et ainsi de suite sauf pour la dernière route qui réunira les villes n+m-1 et $n+\bar{m}$. Le tour est complété en utilisant uniquement des arêtes de coût nul. La figure 3.4 illustre un 3-tour pour un m_ℓ-PVC avec n = 5, $\bar{m}$ = 3 et le tour qui lui correspond par la transformation de Hong et Padberg. Les coûts fixes et les arêtes de coût nul sont explicitement indiqués dans le tour.

FIGURE 3.4

Illustration de la transformation de Hong et Padberg



Les villes $n+\bar{m}+1,\ldots,n+\bar{m}+4$ dans la transformation de Hong et Padberg servent à empêcher que le commis échappe à certains coûts fixes dans le PVC obtenu. Récemment Discenza [32] a montré que les villes $n+\bar{m}+1$ et $n+\bar{m}+4$ ne sont pas nécessaires pour atteindre ce but et peuvent donc être éliminées. Notons en dernier lieu que la transformation de Hong et Padberg peut traiter du cas d'un $\bar{m}$-PVC et aussi celui d'un m_ℓ-PVC avec borne inférieure sur m.

3.2.6 La transformation de Rao [102]

La dernière transformation dont il sera brièvement sujet dans cette section est due à Rao [102]. Il n'est pas nécessaire de la donner en détails car elle est une transposition directe au cas symétrique des équations (3.5) définissant la transformation de Bellmore et Hong. Rao a toutefois le mérite d'avoir réussi à montrer qu'il est valide de procéder ainsi et nous référons le lecteur à [102] pour plus de détails au sujet de la preuve. Cet article de Rao vient d'ailleurs infirmer l'affirmation de Hong et Padberg à savoir que la transformation de Bellmore et Hong ne se transpose pas dans le cas symétrique.

3.3 Formulations des m-PVC asymétriques

3.3.1 Introduction

Tous les modèles présentés dans cette section sont très similaires à ceux exposés en 2.2 concernant les PVC asymétriques. Il y a deux raisons à ce phénomène:

1. pour résoudre le m-PVC(1), on doit prendre le même type de décisions que dans la résolution du PVC et donc les variables respectives de ces deux problèmes sont analogues;
2. dans le PVC tous les sous-tours sont illégaux; dans le m-PVC par contre, certains sous-tours sont permis, notamment ceux contenant le dépôt; il faut donc s'assurer dans le m-PVC que seuls les sous-tours non permis sont éliminés et ceci peut

(1) Dans 3.3, m-PVC signifie m-PVC asymétrique.

s'effectuer en utilisant un sous-ensemble des contraintes d'élimination de sous-tours du PVC, contraintes dont la forme a été vue au chapitre 2.

Les commentaires associés aux formulations présentées ici seront par conséquent assez brefs et le lecteur est référé au chapitre 2 pour les détails manquants.

La plupart des modèles que nous décrirons concernent en fait le m_ℓ-PVC[1]. Pour la majorité d'entre eux, leur forme est peu modifiée si m est fixé à $\bar{m}$. Si tel n'est pas le cas nous donnerons explicitement les modifications appropriées.

Soit G = (N,A) le graphe orienté d'ordre n associé au m-PVC et G' = (N,A') un sous-graphe de G formant un m-tour dans G avec dépôt au sommet 1[2]. Le graphe G' satisfait aux trois propriétés suivantes:

1. le degré intrant et le degré extrant du sommet 1 est égal à m;
2. le degré intrant et le degré extrant des sommets 2,...,n est égal à 1;
3. G' est connexe.

Le modèle général suivant pour le m-PVC reflète ces 3 propriétés d'un m-tour:

$$\text{Min } Z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} + cm \tag{3.7}$$

(P1) assujettie à:

$$\sum_{j=2}^{n} x_{1j} = m \tag{3.8}$$

$$\sum_{i=2}^{n} x_{i1} = m \tag{3.9}$$

(1) Nous supposons dans les modèles présentés que $\bar{m} = n-1$. La contrainte $m \leq \bar{m}$ n'est donc pas nécessaire.

(2) A partir de maintenant, le dépôt sera toujours identifié à ce sommet dans ce chapitre.

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad (i = 2,\ldots,n) \tag{3.10}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad (j = 2,\ldots,n) \tag{3.11}$$

Les x_{ij} ne forment pas de sous-tours illégaux (3.12)

$$x_{ij} = 0,1 \qquad (i,j = 1,\ldots,n) \tag{3.13}$$

$m \geq 1$ et entier (3.14)

Dans (3.7) le coût global d'un m-tour est évalué comme la somme des coûts de transport plus le coût fixe d'utilisation de m commis. Les contraintes (3.8) jusqu'à (3.11) constituent les contraintes de degré pour le m-PVC et sont associées aux deux premières propriétés d'un m-tour.

Soulignons que (3.14) peut simplement s'écrire $m \geq 0$ car:

1. on peut redéfinir m par le nombre de commis en sus de 1 utilisés; le membre droit de (3,8) et (3,9) doit alors s'écrire 2+2m;
2. (3.13) et (3.8) impliquent que m prendra une valeur entière.

Les contraintes de (3.12) assurent la connexité dans un m-tour et servent explicitement à empêcher la formation de sous-tours ne contenant pas le dépôt car les contraintes de degré ne suffisent pas à elles seules à empêcher leur apparition. Nous dénommerons les contraintes de (3.12) les contraintes d'élimination de sous-tours illégaux. On retrouve dans la littérature plusieurs formes pour ces dernières et nous expliciterons les principales ici. Quel que soit le type des contraintes utilisées dans (3.12), on peut soulever le point suivant: puisque par (3.12) tout sous-tour contient la ville 1, (3.8) et (3.9) sont redondantes par rapport à (3.10) et (3.11). Les contraintes (3.8) et (3.9) peuvent donc être omises dans P1[1] pour le m_ℓ-PVC et remplacées par (3.15) ci-après pour le $\bar{m}$-PVC:

$$\sum_{j=2}^{n} x_{1j} = \bar{m} \tag{3.15}$$

---

(1) Dans notre algorithme on verra qu'il est cependant utile de conserver (3.8) et (3.9) dans P1.

Notons qu'il est facile de modifier P1 pour tenir compte de coûts fixes différents: le terme gm est remplacé par $\sum_{i=1}^{\bar{m}} m_i q_i$ dans (3.7). $\bar{m}$ est remplacé par $\sum_{i=1}^{\bar{m}} m_i$ dans (3.8) et (3.14) devient $m_i = 0,1$ $(i = 1,\ldots,\bar{m})$.

3.3.2 Contraintes d'élimination de sous-tours illégaux de Miller et al. [94]

A notre connaissance, les premiers chercheurs à mettre au point une formulation mathématique du m-PVC sont Miller et al. [94]. Leur modèle ne contient pas (3.8) et (3.9). La forme de leurs contraintes d'élimination de sous-tours illégaux a été donnée au chapitre 2 dans l'équation (2.13). Ces contraintes sont valides qu'il s'agisse du PVC ou bien du m-PVC car elles concernent uniquement les sous-tours ne contenant pas le dépôt. Elles peuvent aussi être modifiées pour imposer un nombre maximum de villes sur chacune des routes du m-tour (voir [94]).

3.3.3 Nos contraintes d'élimination de sous-tours illégaux

La forme de (3.12) que nous utilisons dans nos algorithmes pour le m-PVC est une généralisation directe des contraintes de Dantzig et al. [28] données en (2.9). Elles s'écrivent:

$$\sum_{i\in S, j\in S} x_{ij} \leq |S|-1 \qquad (S \subseteq \{2,\ldots,n\}) \quad (3.16)$$

(3.16) opère de façon identique à (2.9) sauf que dans (3.16) on se restreint aux ensembles S dans {2,...,n} tandis que dans (2.9) tout ensemble S strictement inclus dans N fait l'objet d'une contrainte. La raison de ceci est évidemment que, dans le m-PVC, les sous-tours contenant le dépôt sont permis. Notons encore ici l'équivalence entre (3.16) et (3.17) ci-après telle que démontrée dans le théorème 2.1:

$$\sum_{i\in S, j\in \bar{S}} x_{ij} \geq 1 \qquad (S \subseteq \{2,\ldots,n\}) \quad (3.17)$$

3.3.4 Contraintes d'élimination de sous-tours illégaux de Gavish et Graves [45]

Une des formulations mathématiques du m-PVC les plus récentes est due à Gavish et Graves [45]. Encore ici, ce modèle est une généralisation immédiate du modèle que ces mêmes auteurs ont mis au point pour le PVC (voir 2.2.4). La forme des contraintes d'élimination de sous-tours illégaux que Gavish et Graves présentent dans leur article concerne un $\bar{m}$-PVC. Nous les avons adaptées au cas du m_ℓ-PVC en respectant toutefois l'idée originale de leurs auteurs, ce qui nous conduit aux contraintes (3.18) et (3.19) suivantes:

$$\sum_{j=1}^{n} y_{ij} - \sum_{j=1}^{n} y_{ji} = 1 \qquad (i = 2,\ldots,n) \qquad (3.18)$$

$$y_{ij} \leq (n+\bar{m}-2)x_{ij} \qquad (i,j = 1,\ldots,n) \qquad (3.19)$$

où les y_{ij} sont des variables de flot non-négatives associées à chaque arc du graphe G. L'idée de (3.18) est la même que celle de (2.19) cependant ici on exige qu'il y ait accumulation[1] d'une unité de flot à tous les sommets du graphe sauf le sommet-dépôt. Par (3.19) le flot ne peut circuler que sur des arcs utilisés par un commis[2]; donc pour qu'un sous-tour ne contenant le dépôt puisse se former sans violer (3.18) il faudrait que sur les arcs le constituant, le flot augmente d'une unité d'un sommet à l'autre, ce qui est clairement impossible vu la nature cyclique d'un tel sous-tour. Le membre droit de (3.19), notamment $n+\bar{m}-2$, a été choisi en fonction du fait que dans tout $\bar{m}$-tour on compte $n+\bar{m}-1$ arcs. On pourra donc, pour un $\bar{m}$-tour donné, faire circuler du flot sur $n+\bar{m}-1$ variables de flot y_{ij} entre autres dans les $(n+\bar{m}-1)$ quantités $0,1,\ldots,n-\bar{m}-2$ et ceci tout en respectant (3.18) et (3.19).

3.3.5 Contraintes d'élimination de sous-tours illégaux de Svestka et Huckfeldt [119]

Le modèle de Svestka et Huckfeldt décrit ici ne s'applique qu'au

---

(1) Dans (2.19) il s'agit d'une diminution d'une unité de flot.

(2) Un arc (i,j) avec $x_{ij} = 1$.

$\bar{m}$-PVC. Il est construit en transformant tout d'abord le $\bar{m}$-PVC en un PVC par la transformation d'Eilon et al. [35] puis ensuite en formulant le PVC obtenu par un modèle du style de celui de Miller et al. [94]. L'idée de Svestka et Huckfeldt est clairement juste mais son application comporte une fausseté qui a été relevée par Gavish [43].

Supposons que les villes 1,2,...,n-1 sont les villes à visiter et que les villes n,...,n+$\bar{m}$-1 sont les villes-dépôts dans le PVC associé au $\bar{m}$-PVC (avec dépôt en n) par la transformation d'Eilon et al. La contrainte d'élimination de sous-tours illégaux proposée par Svestka et Huckfeldt est:

$$u_i - u_j + \left[\frac{n}{\bar{m}}\right] x_{ij} \leq \left[\frac{n}{\bar{m}}\right] + \bar{m} - 2 \quad (i \leq i \neq j \leq n-1) \quad (3.20)$$

(3.20) n'est pas valide pour $\bar{m} \geq 2$ tel que démontré par Gavish. En effet puisque $x_{ij} = 0,1$, (3.20) prend la forme:

$$u_i - u_j \leq \bar{m} - 2 \qquad \text{si } x_{ij} = 1 \qquad (3.21)$$

$$u_i - u_j \leq \left[\frac{n}{\bar{m}}\right] + \bar{m} - 2 \qquad \text{si } x_{ij} = 0 \qquad (3.22)$$

Le membre droit de (3.21) et (3.22) est toujours non-négatif pour $\bar{m} \geq 2$ et dans ce cas (3.20) sera toujours satisfaite en posant $u_i = 0$ pour tout i. (3.20) n'élimine donc aucun sous-tour.

Gavish [43] propose la contrainte valide suivante dans le cadre de l'idée de Svestka et Huckfeldt:

$$u_i - u_j + (n-\bar{m}) x_{ij} \leq n - \bar{m} - 1 \qquad (1 \leq i \neq j \leq n-1) \qquad (3.23)$$

Il est facile de démontrer que (3.23) élimine tout sous-tour ne contenant pas le dépôt. La preuve est similaire à celle donnée pour (2.13).

Soulevons à la décharge de Svestka et Huckfeldt que cette erreur

n'a pas compromis la validité de l'algorithme qu'ils ont mis au point à partir de leur formulation, tel que Svestka [117] le mentionne dans sa réponse à l'article de Gavish. Nous reviendrons sur cet algorithme en 3.5.

3.4 Formulations des m-PVC symétriques

3.4.1 Introduction

Le nombre de modèles construits pour le m-PVC symétrique semble beaucoup plus restreint que dans le cas asymétrique. Un phénomène similaire se produit pour le PVC et encore ici la raison en est que le graphe associé à un m-PVC symétrique est non-orienté; on ne peut donc plus distinguer l'entrée d'une ville de sa sortie. Ainsi les contraintes de Miller et al. [94] de même que celles de Gavish et Graves [45] ne peuvent plus être utilisées pour éliminer les sous-tours illégaux. Toutefois les contraintes d'élimination de sous-tours illégaux que nous avons décrites en 3.3.3 peuvent s'adapter facilement à un m-PVC symétrique et dans ce sens sont indépendantes de la symétrie ou de l'asymétrie de C. Le modèle ainsi obtenu est décrit en 3.4.2.

3.4.2 Notre modèle pour le m-PVC symétrique

Le modèle que nous avons mis au point pour le m-PVC[1] généralise celui de Dantzig et al. [28] concernant le PVC. Soit G = (N,A) le graphe non-orienté d'ordre n associé au m-PVC et G' = (N,A') le graphe d'un m-tour. G' satisfait aux trois propriétés suivantes:

1. le sommet 1 possède un degré de m;
2. les sommets 2,...,n ont un degré de 1;
3. G' est connexe.

La formulation mathématique suivante est valide pour le m-PVC:

---

(1) Dans 3.4.2, m-PVC signifie m-PVC symétrique.

(P2)

$$\text{Min } Z = \sum_{i<j} c_{ij} x_{ij} + gm \tag{3.24}$$

<u>assujettie à</u>:

$$\sum_{j=2}^{n} x_{1j} = 2m \tag{3.25}$$

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2 \qquad (k = 2,\ldots,n) \tag{3.26}$$

$$\sum_{\substack{i\in S, j\in S \\ i<j}} x_{ij} \leq |S|-1 \qquad (S \subseteq N-\{1\}) \tag{3.27}$$

$$x_{ij} = 0,1 \qquad (2 \leq i<j \leq n) \tag{3.28}$$

$$x_{1j} = 0,1,2 \qquad (j = 2,\ldots,n) \tag{3.29}$$

$$m \geq 1 \text{ et entier} \tag{3.30}$$

Dans (3.24) le coût d'un m-tour est évalué de la façon habituelle[1]. Les contraintes (3.25) et (3.26) constituent les contraintes de degré dans le cas symétrique et correspondent aux deux premières propriétés d'un m-tour. Les contraintes (3.28), (3.29) et (3.30) reflètent les définitions des variables x_{ij} et m. Comme précédemment (3.30) peut s'écrire $m \geq 0$. Soulignons que les variables associées à des arêtes incidentes au dépôt, notamment les variables x_{1j} $(j = 2,\ldots,n)$, peuvent prendre la valeur 2 de façon à modéliser un sous-tour lié au dépôt et contenant 2 villes (un aller-retour). Pour obtenir un modèle complètement binaire, on peut remplacer x_{1j} par les deux variables binaires x_{1j} et x_{j1}.

Les contraintes (3.27) servent à empêcher la formation de sous-tours ne contenant pas le dépôt et ne concernent, pour cette raison, que les sous-ensembles de N n'incluant pas cette ville. Le modus operandi de (3.27) a été vu au chapitre 2 lors de notre discussion au sujet de (2.24) et (2.26). Dans le cas du m-PVC toutefois, la cardinalité de S

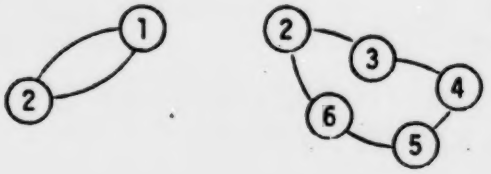
---

(1) On pourrait modifier P2 de façon similaire à P1 pour tenir compte de coûts fixes différents.

dans (3.27) varie de 3 à n-2. Les 2 raisons suivantes justifient cette affirmation. Premièrement tout sous-tour contenant 2 villes différentes du dépôt est implicitement éliminé car les variables x_{ij} avec $i > 1$ ne peuvent prendre que les valeurs 0 ou 1. Deuxièmement, des sous-tours illégaux de cardinalité n-2 peuvent survenir sans la présence de (3.27) comme le démontre la figure 3.5 où une solution irréalisable de P2, satisfaisant les contraintes de degré mais violant une des contraintes de (3.27), est illustrée avec $n = 7$ et $m = 1$.

FIGURE 3.5

Exemple d'un sous-tour illégal contenant n-2 villes



Une forme équivalente de (3.27), tel que vu au chapitre 2 dans le théorème 2.2 est donnée par (3.31):

$$\sum_{\substack{i\in S, j\in \bar{S} \\ \text{ou} \\ i\in \bar{S}, j\in S \\ i<j}} x_{ij} \geq 2 \qquad (S \subseteq N-\{1\}) \qquad (3.31)$$

Soulignons, en terminant nos commentaires au sujet de P2, que si m est libre, (3.25) est redondante[1]. Dans notre algorithme pour le m-PVC où initialement (3.27) est relaxée il est toutefois important d'inclure (3.25) comme on le verra en 3.6.

3.4.3 Le modèle de Gavish et Srikanth [47] pour le m-PVC symétrique

Le modèle de Gavish et Srikanth [47] pour le m-PVC généralise

(1) La preuve est simple et on réfère le lecteur à 3.3.1 où une affirmation similaire est expliquée en détails.

aussi celui de Dantzig et al. dans ce sens que ses contraintes d'élimination de sous-tours illégaux sont de la forme (3.27). Le modèle de Gavish et Srikanth se différencie du nôtre dans les points suivants:

1. des variables x_{1j} et x_{j1} $(j = 2,\ldots,n)$ sont utilisées pour modéliser ce qu'ils appelent des sous-tours immédiats ("allers-retours").

2. La contrainte redondante suivante est ajoutée à P2:

$$\sum_{i<j} x_{ij} = n-1 \tag{3.32}$$

(3.32) est valide car dans tout m-tour on a:

$$\sum_{i<j} x_{ij} + \sum_{j=2}^{n} x_{j1} = n+m-1 \text{ et aussi } \sum_{j=2}^{n} x_{j1} = m.$$

or (3.32) est déduit en faisant la différence entre les deux égalités précédentes.

3. Les contraintes redondantes suivantes sont adjointes à P2 pour $i = 2,\ldots,n$:

$$x_{1i} + x_{i1} - \delta_i \leq 1 \tag{3.33}$$

$$\sum_{i=1}^{n} \delta_i \leq m-1 \tag{3.34}$$

$$\delta_i = 0,1 \tag{3.35}$$

Les contraintes (3.33) jusqu'à (3.35) sont valides car le nombre de sous-tours immédiats est plus petit ou égal à $m-1$ si $m < n-1$.

Gavish et Srikanth ajoutent à P2 les contraintes (3.32) jusqu'à (3.35) dans le double but d'améliorer la qualité de la borne obtenue par relaxation lagrangienne d'une part et d'autre part dans le but de conduire à un problème lagrangien facile à résoudre optimalement. Nous reviendrons sur ce sujet en 3.5 où l'algorithme de Gavish et Srikanth sera décrit brièvement.

3.5 La construction d'algorithmes optimaux pour le m-PVC: commentaires généraux et revue de la littérature

Les modèles décrits en 3.3 et 3.4 pour le m-PVC possèdent la forme générale suivante:

(MG)[1]

Min Z = objectif linéaire (3.36)

assujettie à:

contraintes de degré (3.37)

contraintes d'élimination de sous-tours illégaux (3.38)

contraintes de définitions sur les variables (3.39)

L'objectif à minimiser (eq. 3.36) est une combinaison linéaire des coûts de transport et des coûts fixes d'un m-tour. La forme et le nombre des contraintes de degré (eq. 3.37) varient selon que C est symétrique ou asymétrique. En particulier lorsque C est symétrique, il y a n contraintes de degré sous la forme de (3.25) et (3.26). Lorsque C est asymétrique, on compte 2n contraintes de degré sous la forme de (3.8) jusqu'à (3.11). Le modèle MG contient "environ"[2] 2 fois plus de variables[3], c'est-à-dire [n(n-1)+1], quand C est asymétrique que si C est symétrique où on a $[\frac{1}{2}n(n-1)+1]$[4] variables. Les contraintes d'élimination de sous-tours illégaux peuvent prendre plusieurs formes dans le cas asymétrique, plus précisément (2.13), (3.16), (3.17) ou (3.18-19) tandis que dans le cas symétrique, elles ont la forme de (3.27) ou la forme équivalente (3.31).

Les deux particularités de MG qui le rendent difficile à traiter sont:

1. un très grand nombre de contraintes dans (3.38);

---

(1) MG et certaines remarques qui le suivent peuvent s'appliquer aussi au PVC.

(2) "Exactement", dans le cas d'un $\tilde{m}$-PVC.

(3) On ne compte pas ici les variables introduites dans (3.18-19) et (2.13).

(4) On suppose dans ce nombre que les variables x_{1j} peuvent prendre les valeurs 0,1 ou 2.

2. l'intégrité est exigée sur les variables.

Pour se convaincre du point 1, le tableau 3.1 donne le nombre de contraintes d'élimination de sous-tours illégaux dans MG en fonction de la forme utilisée et du type de la matrice des distances:

TABLEAU 3.1

Nombre de contraintes dans (3.38)
selon leur forme et selon le type de C

CAS ASYMETRIQUE			CAS SYMETRIQUE	
Forme	Nombre		Forme	Nombre
(2.13)	$(n-1)(n-2)$		(3.27) ou (3.31)	$\binom{n-1}{3}+\ldots+\binom{n-1}{n-2}$
(3.16) ou (3.17)	$2\left[\binom{n-1}{2}+\ldots+\binom{n-1}{\frac{1}{2}(n-2)}\right]+\binom{n-1}{n-2}$	si n pair		
	$2\left[\binom{n-1}{2}+\ldots+\binom{n-1}{\frac{1}{2}(n-3)}\right]+\binom{n-1}{\frac{1}{2}(n-1)}+\binom{n-1}{n-2}$	si n impair		
(3.18) (3.19)	$(n+1)(n-1)$			

D'après le tableau 3.1, il est évident que (3.16) ou (3.17) de même que leur contrepartie symétrique, c'est-à-dire (3.27) ou (3.31), conduisent à un nombre très élevé de contraintes d'élimination de sous-tours illégaux, nombre qui est d'ailleurs exponentiel en n. Pour le cas asymétrique, le nombre de contraintes du type (2.13) ou de (3.18-19) est proportionnel à n^2 et est donc beaucoup moins élevé que le nombre de contraintes du type (3.16) ou (3.17). Leur utilisation dans MG conduit toutefois à un modèle en nombre entiers encore beaucoup trop grand pour être résolu directement en plus du fait qu'elles impliquent l'ajout de variables supplémentaires[1] au modèle.

Pour pallier aux deux difficultés mentionnées précédemment, les

(1) Les variables u_i ou les variables y_{ij}.

algorithmes décrits dans la littérature , y compris le nôtre, sont basés sur le choix d'une relaxation de MG. Certains auteurs relaxent de façon lagrangienne les contraintes de degré aux noeuds autres que le dépôt dans MG pour que le calcul d'une solution optimale avec les contraintes non relaxées puisse s'effectuer par la recherche d'un m-arbre (cas symétrique) ou d'une m-arborescence (cas asymétrique) de poids minimal. Les avantages d'une telle procédure sont doubles: le problème d'optimisation du lagrangien est P-complet et l'intégrité des variables est atteinte automatiquement. Les algorithmes de Ali et Kennington [1] et de Gavish et Srikanth [47], extensions directes des travaux de Held et Karp [60,61] au PVC, procèdent de cette façon. Gavish et Srikanth, toutefois, en plus des contraintes de degré aux sommets 2,...,n qui sont introduites dans l'objectif avec un certain poids dual, ajoutent aussi dans l'objectif de façon lagrangienne les contraintes de (3.33). Le lagrangien à résoudre est donc différent de celui utilisé par Ali et Kennington. Dans les deux algorithmes mentionnés précédemment la valeur des variables duales est calculée par une technique de sous-gradients [109].

D'autres auteurs conservent les contraintes de degré et relaxent les contraintes (3.38); ce qui conduit à un problème de flot à coût minimum dans le cas asymétrique et à un problème de "b-matching" [33] dans le cas symétrique. Jusqu'à présent aucun chercheur n'a utilisé une telle relaxation de façon lagrangienne[(1)]. L'algorithme de Svestka et Huckfeldt [119], extension des travaux de Bellmore et Malone [7] sur le PVC, utilise une méthode de "Branch-and-Bound" où on tient compte implicitement de (3.38) au niveau du branchement[(2)]. L'intégrité est maintenue automatiquement car l'algorithme de Svestka et Huckfeldt traite d'un m-PVC asymétrique où, lorsque seules les contraintes de degré sont présentes, le problème est unimodulaire.

Le principe général sur lequel sont basés nos deux algorithmes pour le m-PVC consiste en une relaxation complète initialement des

---

(1) Pour faire suite aux travaux de Balas et Christofides [2] sur le PVC asymétrique.

(2) Ce qui explique pourquoi leur algorithme n'est pas invalidé par la fausseté de leur modèle pour le m-PVC.

contraintes d'élimination de sous-tours illégaux et des contraintes d'intégrité sur les variables. A l'inverse de l'algorithme de Svestka et Huckfeldt cependant, lorsqu'on découvre qu'une ou plusieurs des contraintes de (3.38) sont violées par la solution optimale de notre relaxation, on réintroduit explicitement celles-ci[1] dans le modèle. On obtient une nouvelle relaxation plus forte que la précédente mais plus difficile à résoudre optimalement vu que l'intégrité doit être maintenue à l'aide de coupes de Gomory ou à l'aide d'une méthode de "Branch-and-Bound".

3.6 Deux nouveaux algorithmes pour le m-PVC

3.6.1 Introduction

Cette section contient la description de deux algorithmes optimaux que nous avons mis au point pour le m-PVC. La description en sera volontairement très détaillée car les méthodes que nous décrirons dans les chapitres subséquents sont basées sur le même principe. Nous pourrons donc dans ceux-ci omettre certains détails et référer le lecteur à 3.6.

Nous appellerons ces deux algorithmes l'algorithme NORMAL et l'algorithme RENVERSE. Nous donnerons tout d'abord la description générale et quelques commentaires sur la relaxation utilisée dans ces deux algorithmes. Suivra une discussion sur certains points importants pouvant affecter la performance des méthodes qu'on peut déduire à partir d'une telle relaxation. Finalement une description plus technique des algorithmes NORMAL et RENVERSE sera effectuée.

3.6.2 Description générale de nos deux algorithmes

Nos algorithmes pour le m-PVC opèrent en général de la façon suivante:

Etape 1:
Résoudre par la méthode du simplex la relaxation linéaire obtenue en

---

(1) On verra en 3.6 quelle forme de (3.38) a été choisie.

éliminant de MG les contraintes (3.38) et en ne conservant des contraintes d'intégrité (3.39) que les bornes supérieures qu'elles impliquent sur les variables. Le modèle résolu s'appelle MA et peut se décrire par:

(MA)
- Min objectif
- <u>assujettie à</u>:
- contraintes de degré
- $0 \leq x_{ij}^{m} \leq b_{sup}$

Dans ce qui suit, MA^+ désigne le modèle MA auquel est adjoint un certain nombre de contraintes de (3.38). MA est donc un cas particulier de MA^+.

<u>Etape 2</u>:
Si la solution optimale de MA^+ viole une des contraintes de (3.38) ou (3.39), on lui ajoute des contraintes qui rendent irréalisable la solution optimale présente mais qui sont valides pour le m-PVC puis on va à l'étape 3. Sinon, on termine.

<u>Etape 3</u>:
On réoptimise MA^+ par des itérations duales du simplex puis on retourne à l'étape 2. ■

Les contraintes ajoutées à l'étape 2 sont, soit des coupes de Gomory pour restaurer l'intégrité, soit des contraintes de (3.38) pour éliminer des sous-tours illégaux présents dans la solution optimale de MA^+. Une telle procédure ne peut donc en théorie se terminer que par une des deux façons suivantes:

1. toutes les contraintes de (3.38) ont été ajoutées au problème MA original ainsi que suffisamment de coupes de Gomory pour que la solution optimale soit entière;

2. on obtient à l'étape 2 une solution entière ne contenant aucun sous-tour illégal, même si MA^+ ne contient pas toutes

les contraintes de (3.38). Cette solution est optimale pour MG car elle satisfait aussi les contraintes de (3.38) non incluses dans MA^+.

Théoriquement, nous obtiendrons donc toujours la solution optimale de MG par une telle procédure. Il est clair a priori que l'efficacité de celle-ci sur un m-PVC donné dépend des points suivants:

1. la facilité avec laquelle on peut résoudre et réoptimiser MA^+;
2. la rapidité avec laquelle on peut découvrir qu'une solution contient des sous-tours illégaux et adjoindre les contraintes de (3.38) appropriées;
3. la nombre de coupes de Gomory qu'on devra engendrer;
4. le nombre de contraintes de (3.38) que l'on devra engendrer.

Le point 1 ne constitue pas vraiment une difficulté majeure. La méthode du simplex peut résoudre assez facilement, même pour n grand, le problème MA original qui est de la forme d'un problème de flot à coût minimum [37] dans le cas asymétrique et de la forme d'un problème de "b-matching" [33] dans le cas symétrique. Soulignons qu'il n'est pas nécessaire de résoudre ce problème par l'algorithme le plus efficace car d'autres contraintes lui seront adjointes et sa structure initiale sera perdue. La méthode du simplex s'est avérée efficace pour résoudre des problèmes linéaires et, à moins que la taille de MA^+ devienne très grande, on peut s'attendre à des optimisations rapides des problèmes rencontrés.

Le point 2 ne s'avère pas, lui non plus, un obstacle difficile à surmonter. La découverte de sous-tours illégaux s'effectue par un processus d'étiquetage[1] identifiant les composantes connexes du graphe associé à une solution optimale de MA^+. Soit $G' = (N', A')$ ce graphe et $\bar{X} = (\bar{x}_{ij})$ la solution analysée. On a:

---

(1) Nous avons modifié le processus d'étiquetage mis au point par Miliotis [90] pour le PVC.

$N' = N$

$A' = \{(i,j) \in A \mid \bar{x}_{ij} > 0\}$

Par définition d'une composante connexe, les sommets i et j font partie de la même composante connexe de G' s'il existe une chaîne ou un chemin[1] les reliant dans G'. Au début du processus d'étiquetage, tous les sommets de N se voient accorder une étiquette différente comme si chacun faisait partie d'une composante connexe différente. Par la suite, chaque variable x_{ij} est analysée exactement une fois et si $\bar{x}_{ij} > 0$ alors il existe une chaîne ou un chemin reliant i,j et tous les sommets possédant la même étiquette que i ou j. Ces sommets font donc tous partie de la même composante connexe et ils recevront, disons, l'étiquette de i.

Une fois les valeurs de toutes les variables x_{ij} examinées, ou bien tous les sommets ont la même étiquette et G' ne contient qu'une seule composante connexe[2], ou bien certains sommets ont des étiquettes distinctes, ce qui indique la présence d'un ou plusieurs sous-tours illégaux.

Si S est un ensemble de sommets ayant la même étiquette à la fin du processus d'étiquetage et ne contenant pas le dépôt, alors une des composantes de G' constitue un sous-tour sur S, sous-tour qui viole une des contraintes de (3.38). La structure du code de Land et Powell [71] rend facile l'ajout de cette contrainte (voir 1.6).

Le nombre d'opérations nécessaires pour compléter le processus d'étiquetage précédemment décrit est proportionnel à n^2 car chaque variable x_{ij} est analysée exactement une fois.

Le point 3 concerne la facilité avec laquelle on pourra restaurer l'intégrité dans la solution de MA^+. Si un très grand nombre de coupes

---

(1) Selon que G' est symétrique ou asymétrique.

(2) Donc G' est connexe par définition.

de Gomory est nécessaire, beaucoup de temps d'ordinateur sera utilisé pour les calculer, les adjoindre à MA^+ et pour réoptimiser. De plus, ces contraintes ont une densité de coefficients nuls très petite en moyenne et elles accaparent donc beaucoup d'espace-mémoire. Dans un m-PVC asymétrique avec n = 100, chaque coupe de Gomory adjointe à MA^+ pourra possiblement impliquer l'occupation de 9901 mots-mémoire(1).

Le point 4 a rapport avec la proximité de notre relaxation initiale par rapport à MG. Si MA^+ s'avère loin(2) de MG, un grand nombre de contraintes de (3.38) devront être ajoutées. Il s'ensuivra alors un grand nombre de réoptimisations et l'utilisation de beaucoup d'espace-mémoire, quoique ce désavantage est moins important avec les contraintes d'élimination de sous-tours illégaux qu'avec les coupes de Gomory.

A la lumière des remarques précédentes, on peut affirmer que le succès ou l'échec d'une telle procédure sur un m-PVC donné dépend en grande partie du nombre de coupes de Gomory et du nombre de contraintes de (3.38) que l'on devra engendrer pour le résoudre. De plus, si l'on ajoute un grand nombre de telles contraintes, l'espace-mémoire nécessaire pour effectuer cette procédure sur ordinateur et son temps d'exécution seront d'autant plus grands. On verra en 3.6.5 par les tests de performance que nous avons effectués que c'est surtout le facteur mémoire qui est pénalisé par l'ajout d'un grand nombre de contraintes.

La procédure précédente est une extension au m-PVC des travaux de plusieurs chercheurs concernant le PVC(3): Dantzig et al. [28], Martin [89], Miliotis [92], Grötshel [54], Land [69], Padberg et Honn [98] ainsi que Crowder et Padberg [27]. Quelques-uns de ceux-ci ont pu résoudre

---

(1) Dans un tel m-PVC, il y a (100 x 99)+1 variables; on ne tient pas compte ici du signe, du membre droit et de l'écart sur cette contrainte qui occuperont aussi de l'espace-mémoire.

(2) On peut prendre comme mesure de proximité entre une relaxation et le problème original, la différence en valeur absolue entre le coût de la solution optimale de la relaxation et le coût de la solution optimale du problème original.

(3) Voir chapitre 2, section 2.6.2.

des PVC de plus de 100 villes en n'utilisant qu'un nombre très restreint de contraintes dont des coupes de Gomory, des contraintes d'élimination de sous-tours illégaux et quelques autres types de contraintes valides pour le PVC. Peut-on espérer qu'un tel phénomène se produira aussi lors de l'application de la procédure précédente au m-PVC? Intuitivement il semble que oui. Lorsque m est libre, le m-PVC est une relaxation du PVC car l'ensemble des m-tours inclut celui des 1-tour. Donc notre algorithme fournira toujours une solution légale au m_ℓ-PVC avant ou en même temps qu'une solution légale pour le PVC, c'est-à-dire un tour[1].

Le $\bar{m}$-PVC ($\bar{m} \geq 2$) n'est pas une relaxation du PVC et le raisonnement précédent ne s'applique plus. On verra toutefois, à l'aide des tests sur ordinateur que nous avons effectués, que le $\bar{m}$-PVC est encore plus facile à résoudre que le m_ℓ-PVC ou même le 1-PVC qui est équivalent au PVC. Ce phénomène semble de plus être indépendant de la procédure utilisée car les résultats qu'obtiennent Gavish et Srikanth [47] avec une méthode tout à fait différente pour résoudre le $\bar{m}$-PVC corroborent cette affirmation. Nous donnerons une explication intuitive de ceci plus tard.

Nous avons volontairement décrit certains passages de notre méthode de résolution du m-PVC donnée en page 94 de façon ambigüe dans le but de laisser à un niveau le plus général possible notre discussion au sujet de cette méthode. Dans ce qui suit nous expliquerons de façon complète tous les détails techniques qui nous ont permis de l'implanter sur ordinateur.

3.6.3 Description technique de nos deux algorithmes

Choix d'une relaxation initiale

Premièrement, pour l'étape 1, mentionnons qu'il n'est pas nécessaire dans le cadre d'une telle procédure de définir notre relaxation

---

(1) La première solution réalisable du m_ℓ-PVC obtenue contiendra un ou plusieurs tours. Dans ce dernier cas, il ne s'agit pas d'une solution réalisable pour le PVC.

initiale, c'est-à-dire MA, en y incluant les contraintes de degré[1]. On aurait pu tout aussi bien relaxer celles-ci initialement et les adjoindre par la suite à MA de la même façon que les coupes de Gomory et les contraintes de (3.38). Une telle relaxation extrémale s'avérerait toutefois une perte de temps car dans la solution optimale d'un m-PVC elles sont toujours presque toutes effectives, donc on aurait dû les adjoindre à MA de toutes façons.

Choix d'une méthode d'hyperplans de coupe

A l'étape 2 de la procédure il est implicite que nous avons choisi a priori une méthode d'hyperplans de coupe plutôt qu'une méthode de "Branch-and-Bound" pour restaurer l'intégrité. La raison de ce choix est que, d'après notre expérience[2], une méthode de coupes est plus rapide mais numériquement moins stable qu'une méthode de "Branch-and-Bound". Une méthode de coupes est une méthode duale où la première solution réalisable rencontrée est optimale. Il arrive cependant parfois que les coupes ne convergent pas car elles coupent une partie de plus en plus infime du polytope associé à la relaxation linéaire et des problèmes numériques surviennent alors sur ordinateur, le déterminant de la matrice de base devenant de plus en plus grand. Avec une méthode de "Branch-and-Bound" on ne rencontre pas ce type de problème et même si, faute de temps, la méthode n'est pas complétée, il peut se produire qu'on obtienne quand même une bonne solution réalisable du problème. La structure de données à manipuler avec ce type de méthode est toutefois plus lourde et en général son temps d'exécution est plus élevé qu'avec des coupes si celles-ci convergent. Nous avons donc décidé a priori de travailler avec des coupes et a posteriori nous avons constaté que la difficulté sous-jacente aux coupes n'a été rencontrée que très peu souvent pour le m-PVC.

---

(1) On pourrait débuter avec MA ne contenant qu'une seule contrainte comme par exemple: $\sum_{i<j} x_{ij} = n+m-1$ (cas symétrique).

(2) Basée entre autres sur les études que nous avons effectuées pour les 4 extensions du PVC traitées.

Choix des contraintes d'élimination de sous-tours illégaux: leur forme et leur nombre

Toujours à l'étape 2 de la méthode décrite en page 94 il n'est pas spécifié quelle forme de contraintes d'élimination de sous-tours sera utilisée et combien seront ajoutées à MA^+. Ce choix est primordial car l'ajout de contraintes trop faibles, bien qu'éliminant la solution présente, conduirait à une convergence lente de notre méthode vers la solution optimale du m-PVC. Dans le cas symétrique, nous n'avons pas eu de choix à effectuer car une seule forme de (3.38) nous était connue, notamment (3.27). Nous pouvons utiliser trois formes possibles pour les contraintes (3.38) dans le cas asymétrique. La forme (3.18-19) due à Gavish et Graves [45] ne semble pas convenir car elle introduit trop de variables(1) supplémentaires dans le modèle, ce qui constitue un désavantage majeur dans le type de méthode que nous employons comme nous le verrons plus tard. La forme due à Miller et al. [94], notamment (2.13), implique elle aussi l'ajout dans le modèle de variables(2) supplémentaires mais en plus petit nombre que les contraintes de Gavish et Graves. Nous les avons toutefois rejetées car elles sont moins fortes que les contraintes (3.16) ou les contraintes équivalentes (3.17) comme on l'a montré en 2.2.3.

Les remarques précédentes expliquent pourquoi nous avons choisi d'utiliser pour (3.38) les contraintes du type de celles de Dantzig et al., notamment (3.16) ou (3.17) pour C asymétrique et (3.27) ou (3.31), pour C symétrique. Bien que (3.16) soit mathématiquement équivalent à (3.17), de même que (3.27) est équivalent à (3.31), un détail pratique au niveau de leur implantation sur ordinateur nous permet de distinguer une forme plutôt que l'autre. Considérons tout d'abord le cas asymétrique et dénotons par p la cardinalité de l'ensemble $S \subseteq N-\{1\}$ sur lequel est basée la contrainte de (3.16) et de (3.17) que nous voulons comparer. La contrainte de (3.16) considérée contient $p_1 = p(p-1)$ coeffi-

(1) Notamment les variables de flot y_{ij} dont le nombre est $n(n-1)$.

(2) Les variables u_i dont le nombre est $n-1$.

cients tandis que la contrainte équivalente de (3.17) en contient $p_2 = p(n-p)$. On aura donc $p_1 \leq p_2$ lorsque $p \leq \left\lceil \frac{n+1}{2} \right\rceil$. Il est clair que dans une méthode comme la nôtre, le facteur mémoire joue un rôle primordial et le critère nombre de coefficients induits par une contrainte peut nous servir pour la choisir de préférence à une autre. En particulier (3.16) sera donc préférable à (3.17) selon ce critère pour l'élimination d'un "petit"(1) sous-tour. Dans le cas symétrique une analyse similaire peut être effectuée. Avec la même notation on obtient qu'une contrainte de (3.27) sur S de cardinalité p contient $p_3 = \frac{1}{2}p(p-1)$ coefficients tandis que la contrainte équivalente de (3.31) en contiendra $p_4 = p(n-p)$. Puisque $p_3 \leq p_4$ ssi $p \leq \left\lceil \frac{2n+1}{3} \right\rceil$ on obtient une conclusion similaire au cas asymétrique.

L'analyse précédente nous permet de décider quelle forme de (3.38) il est préférable de choisir. Il nous reste maintenant à décider combien de contraintes de (3.38) il est préférable d'ajouter à MA^+ à chaque passage à l'étape 2 de notre procédure. Une solution optimale de MA^+ peut contenir plus d'un sous-tour illégal et nous sommes placés devant le choix d'adjoindre une ou plusieurs contraintes de (3.38) à MA^+ même si une suffit à rendre irréalisable cette solution. Le dilemme à trancher se pose dans les termes suivants. Ajouter une seule contrainte de (3.38) nous permet d'obtenir l'effet désiré et utilise moins d'espace-mémoire qu'en adjoindre plus d'une. D'un autre côté, n'ajouter qu'une seule contrainte permet une réoptimisation où au moins deux sous-tours sont éliminés et d'autres restent intacts. Nous avons choisi comme première tentative d'ajouter à MA^+ une contrainte par sous-tour illégal dans sa solution optimale de façon à tous les briser. L'impact de ce choix au niveau de l'espace-mémoire utilisée est minime car les contraintes de (3.38) contiennent relativement peu de coefficients par rapport, disons, aux coupes de Gomory. Les résultats sur ordinateur confirment la justesse de cette décision car, comme on le verra, un nombre très restreint de contraintes d'élimination de sous-tours illégaux sont engendrées même avec une politique aussi extrémale. Nous avons choisi une forme unique pour ces contraintes, notamment (3.16) dans le cas asymétrique et (3.27) pour

---

(1) En termes du nombre de villes qu'il contient par rapport à n.

le cas symétrique. Il est clair qu'il pourra survenir une situation où la solution optimale de MA^+ contiendra plusieurs "petits" sous-tours illégaux et quelques autres plus "grands". Pour ces derniers (3.17) ou (3.31) serait préférable. En moyenne cependant ce genre de situation se présente relativement peu souvent et comme le nombre de sous-tours illégaux est très petit il n'y aurait pas un net avantage à varier d'une forme à l'autre selon le type de sous-tour.

En quel nombre ajouter les coupes de Gomory à MA^+ et lesquelles choisir?

On peut se poser des questions similaires pour les coupes de Gomory: en quel nombre et lesquelles doit-on adjoindre à MA^+? En effet, pour chaque variable prenant une valeur non entière dans la solution de MA^+, un et même plusieurs hyperplans de coupe peuvent être ajoutés. Puisque de telles contraintes nécessitent l'utilisation de beaucoup de mots-mémoire nous avons décidé d'adjoindre à MA^+ une seule coupe de Gomory. En outre, lorsque plus tard dans le processus on se rend compte qu'une ou plusieurs de ces coupes deviennent ineffectives, celles-ci sont éliminées; le désavantage évident de procéder ainsi étant qu'il sera peut-être nécessaire de les réintroduire à des étapes subséquentes. On y perd en temps d'exécution mais on y gagne en espace-mémoire, ce qui est beaucoup plus important dans notre procédure. La forme des hyperplans de coupe analysés et le choix de celui qui devra être engendré se fait dans la sous-routine INTCON que nous avons construite en modifiant la sous-routine du même nom mise au point par Miliotis [90]. Cette dernière a été décrite en 2.6.2.

Dans quel ordre obtient-on l'intégrité et la connexité du graphe?

Afin que notre algorithme pour le m-PVC soit complètement spécifié il faut discuter d'une dernière ambiguïté qui subsiste à l'étape 2 de notre algorithme. Il s'agit de l'ordre respectif dans lequel les hyperplans de

coupe et les contraintes de (3.38) seront adjoints à MA^+. Il existe plusieurs possibilités:

1. on peut ajouter uniquement des hyperplans de coupe jusqu'à ce qu'on obtienne une solution entière optimale pour MA^+ puis ensuite vérifier s'il existe des sous-tours illégaux et dans l'affirmative adjoindre des contraintes de (3.38) à MA^+. Si l'intégrité est perdue par l'ajout de ces contraintes, l'intégrité devra être restaurée avant d'analyser les sous-tours obtenus. En résumé, par cette politique, on atteint l'intégrité dans la solution de MA^+ avant de vérifier sa légalité au sens des sous-tours qu'elle contient et l'algorithme qu'elle induit constitue l'algorithme NORMAL;

2. on peut ajouter à MA^+ uniquement des contraintes de (3.38) tant que la solution optimale contient des sous-tours illégaux puis dans la négative vérifier l'intégrité de cette solution. Cette politique est l'inverse de la précédente et conduit à l'algorithme dit RENVERSE. Evidemment, ici le concept de sous-tour doit être élargi à une solution non entière. L'extension se fait simplement: on considère que deux villes, disons i et j, sont reliées par le passage d'un commis de i à j si $x_{ij} > 0$[1];

3. on peut ajouter simultanément à MA^+ des hyperplans de coupe et des contraintes de (3.38) lorsque sa solution optimale est non entière et contient des sous-tours illégaux. Une telle façon de procéder est une politique mitoyenne par rapport aux deux précédentes. Nous n'avons cependant pas testé cette politique.

3.6.4 Implantation sur ordinateur de NORMAL et RENVERSE

Les principales modifications à apporter au programme MIF (section 1.6)

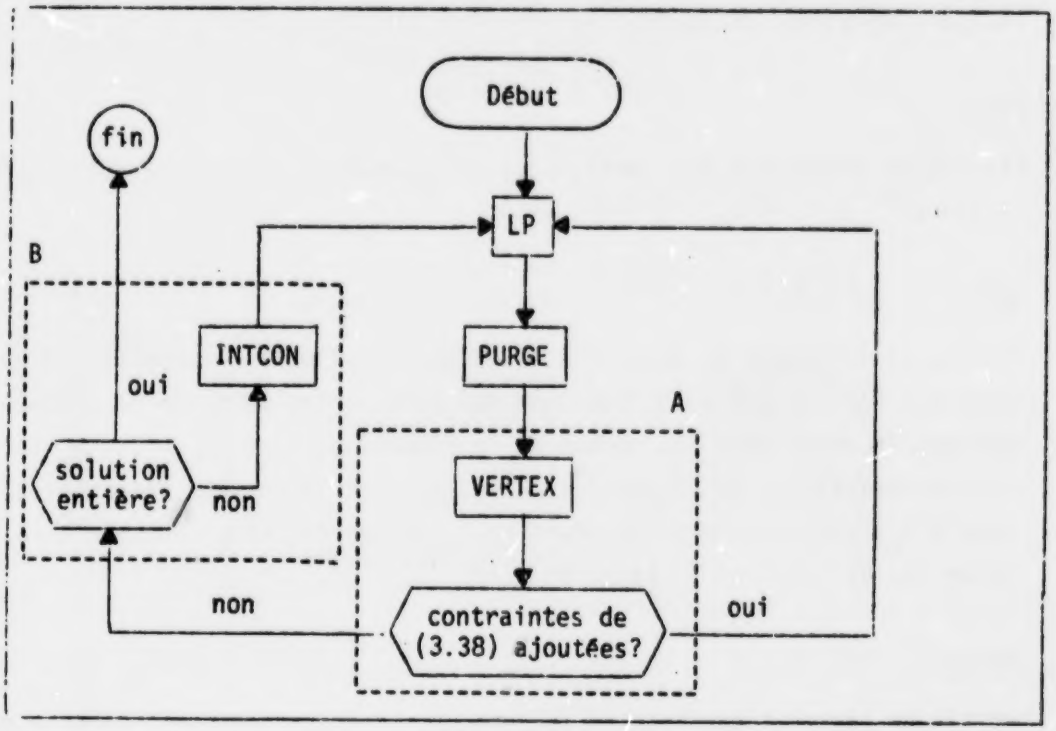
---

(1) $x_{ji} > 0$ si $j < i$ dans le cas symétrique.

pour implanter sur ordinateur les algorithmes NORMAL et RENVERSE ont été décrites en 1.6.2[1]. La figure 1.5 de cette dernière section illustre de façon succincte le fonctionnement de NORMAL. L'algorithme RENVERSE peut être décrit ainsi:

FIGURE 3.6

Algorithme RENVERSE



En intervertissant les parties A et B de l'organigramme de la figure 3.6, on obtient l'organigramme de l'algorithme NORMAL et une adaptation de la figure 1.5 au cas du m-PVC.

Le rôle des sous-routines mentionnées dans la figure 3.6 a été

(1) Dans un contexte plus général toutefois.

décrit en 1.6.1 pour un problème linéaire en nombres entiers quelconque. Nous allons préciser le travail de chacune d'entre elles dans le cas du m-PVC en fonction du matériel présenté en 3.6.3.

LP

Résout initialement la relaxation continue du problème linéaire contenant uniquement les contraintes de degré. A une étape quelconque de l'algorithme, LP résoudra le problème initial auquel est adjoint des hyperplans de coupe et des contraintes de (3.38).

PURGE

Elimine de la matrice des contraintes les coupes de Gomory devenues ineffectives[1].

VERTEX

Vérifie si le graphe G' associé à la solution optimale obtenue par LP est connexe. Dans le cas où G' contient des composantes connexes ne contenant pas le dépôt dont les ensembles de sommets sont notés $S_1,\ldots,S_q$ $(q \geq 1)$ alors q contraintes de (3.38), associées respectivement aux ensembles S_1 jusqu'à S_q, sont adjointes au problème. Chacune de ces q contraintes est violée par la solution linéaire en main.

INTCON

Vérifie si la solution linéaire fournie par LP est entière et, dans la négative, ajoute au problème une coupe de Gomory. Nous avons modifié la sous-routine INTCON de Miliotis [90] de façon à ce qu'une seule coupe soit analysée pour chaque variable fractionnaire. Le même critère de profondeur (voir 2.6.2) sert pour trouver la meilleure.

3.6.5 Tests de performance

Tests où m est variable

Le tableau 3.2 résume les résultats de nos tests pour les algorithmes

(1) Les contraintes de (3.38) sont toujours gardées.

TABLEAU 3.2

Comparaison entre l'algorithme NORMAL et l'algorithme RENVERSE

		ALGORITHME NORMAL							ALGORITHME RENVERSE						
TYPE DE PROBLEME	n	Nombre de problèmes (5)	Taille maximum de l'inverse (6)	Nombre total de contraintes (6)	Nombre de coupes (6)	Nombre de contraintes de (3.38) (6)	Nombre d'itérations du simplex (6)	Temps (sec.) (6)	Nombre de problèmes (5)	Taille maximum de l'inverse (6)	Nombre total de contraintes (6)	Nombre de coupes (6)	Nombre de contraintes de (3.38) (6)	Nombre d'itérations du simplex (6)	Temps (sec.) (6)
SYMETRIQUE EUCLIDIEN (1,2)	50	4	65	70.5	5.5	15	266	81.75	5	65	70.6	5.6	15	255	74.8
	60	5	78	84.4	7	17.4	285	116.8	5	75	78.4	4.6	13.8	202	66.0
	70	3	86	92	5	17	319	162.3	4	86	90.8	2.3	21	256	92.5
	80	-	-	-	-	-	-	-	2	103	111	3.5	27.5	397	283.0
SYMETRIQUE NON-EUCLIDIEN (3)	50	5	54	57	3.4	3.6	159	34.0	5	55	57.2	4	3.2	154	30.0
	60	4	60	62.8	0.5	2.3	161	38.5	5	62	65	1.6	3.4	173	45.2
	70	4	72	74.8	2.0	2.8	204	74.0	5	74	78	3.2	4.8	258	150.8
	80	5	82	84.8	2.2	2.6	445	221.8	5	83	86.4	2.6	3.8	277	137.0
	90	5	94	99.8	3.4	6.4	380	271.4	5	93	97	2.8	4.2	315	184.6
	100	-	-	-	-	-	-	-	3	100	103.3	0.6	2.6	309	208.3
ASYMETRIQUE (4)	50	-	-	-	-	-	-	-	5	102	106.6	2.2	4.4	210	97.6
	60	-	-	-	-	-	-	-	4	122	125.5	1.3	4.3	226	127.7
	70	-	-	-	-	-	-	-	5	141	146.2	1.0	5.2	264	202.6

(1) Quand C est complète et m est libre, le m$_1$-PVC est équivalent au PVC.
(2) Aucun test n'a été effectué avec n = 80 et l'algorithme normal.
(3) Aucun test n'a été effectué avec n = 100 et l'algorithme renversé.
(4) Les problèmes ont été résolus uniquement par l'algorithme renversé.
(5) Nombre de problèmes résolus sur 5.
(6) Valeur moyenne.

NORMAL et RENVERSE. Deux ensembles de problèmes avec 50 villes ou plus ont été utilisés pour comparer ces deux algorithmes, l'un dans le cas symétrique euclidien et l'autre dars le cas symétrique non-euclidien. Seul l'algorithme RENVERSE a été testé sur des problèmes asymétriques. Dans tous les problèmes m est variable et les coûts fixes des commis sont nuls.

Dans le tableau 3.2, la colonne taille maximum de l'inverse donne la dimension maximum qu'a atteint l'inverse réduit de la base (voir section 1.6.1) au cours de l'optimisation. C'est une donnée importante car beaucoup d'espace-mémoire est nécessaire pour conserver cet inverse et par conséquent cette statistique aura une grande influence sur la taille des problèmes qu'on pourra résoudre. Les colonnes "Nombres de coupes" et "Nombre de contraintes de (3.38)" nous fournissent le nombre moyen de contraintes de chaque type qu'il a fallu engendrer pour obtenir la solution optimale. La colonne "Nombre total de contraintes" est le nombre moyen total qu'on a dû engendrer au cours du processus d'optimisation.

Les constatations suivantes se dégagent du tableau 3.2. En ce qui concerne les problèmes symétriques euclidiens en moyenne 5 coupes de Gomory ont suffi pour atteindre la solution optimale. On peut ajouter que, dans la plupart des problèmes résolus, 10 coupes de Gomory ou moins ont suffi. Le nombre de contraintes d'élimination de sous-tours illégaux engendrées augmente évidemment avec n quoique pour tous les problèmes tests, ce nombre n'a jamais dépassé 32. En général, l'algorithme RENVERSE engendre moins de contraintes que NORMAL, par conséquent les temps de résolution obtenus sont moins élevés et des problèmes de plus grande taille peuvent être résolus par cet algorithme.

Les problèmes symétriques non-euclidiens sont en général beaucoup plus faciles à résoudre que les problèmes euclidiens et encore ici l'algorithme RENVERSE est supérieur. Dans tous les problèmes tests résolus, 6 hyperplans de coupe ou moins ont été engendrés et pour ces problèmes une moyenne de 3 ont suffi. Le nombre de contraintes de (3.38) créées restent dans la plupart des cas inférieur à 8 et excède très rarement

10. Ce nombre semble d'ailleurs peu influencé par n et prend une valeur moyenne de 3. Un des problèmes de 100 villes a été résolu sans qu'aucune coupe ni aucune contrainte de (3.38) ne soit engendrée. Comme prévu, m prend la valeur 1 dans la plupart des problèmes sauf dans quelques rares exceptions où m est égal à 2 ou 3.

Pour les problèmes asymétriques, le nombre de contraintes initiales dans MA de même que le nombre de variables est environ le double de celui des problèmes symétriques, ce qui conduit à des temps de résolution beaucoup plus grands car le temps de résolution de la méthode du simplex est très influencé par ces deux quantités. En particulier le nombre de secondes par itération est plus grand que dans le cas symétrique. Le problème de plus grande taille résolu contient 70 villes. Mentionnons toutefois que très peu de coupes de Gomory sont engendrées pour ce type de problème (généralement 0,1 ou 2) et que le nombre de contraintes d'élimination de sous-tours illégaux créées est moins grand en moyenne que pour le cas symétrique. La valeur de m est toujours égale à 1 sauf dans 2 problèmes où m a pris la valeur 2.

Il est clair d'après ces tests que la relaxation que nous avons choisie est très efficace quel que soit le type de problèmes considéré et que la meilleure façon d'effectuer cette relaxation est celle de l'algorithme RENVERSE. En comparant les nombres de la colonne "Taille maximum de l'inverse" à ceux de la colonne "Nombre total de contraintes" on se rend compte que la plupart des contraintes engendrées sont effectives dans la solution optimale d'un problème donné, on peut donc conclure que très peu d'entre elles ont été engendrées inutilement pour l'obtention de cette solution optimale.

Le facteur espace-mémoire est dans tous les cas la cause de l'échec de notre procédure sur un problème-test donné et habituellement un tel phénomène se produit lorsqu'un grand nombre de coupes de Gomory est nécessaire, ce qui implique soit l'obligation d'utiliser un trop grand nombre de mots-mémoire, soit, dans certains cas plus rares, un déterminant de la base très grand[1].

---

(1) Valeur maximale permise: 10,000.

Il nous est possible d'évaluer avec précision la taille mémoire requise par nos deux algorithmes étant donnée la valeur des paramètres suivants:

n : nombre de villes[1];

v : nombre de variables[1];

c : nombre maximal de contraintes qu'il est permis d'adjoindre à MA[2];

p : dimension maximum que l'inverse réduit de la base peut atteindre[2];

a : nombre maximum de coefficients que la matrice des contraintes peut recevoir[2];

t : nombre total de mots-mémoire nécessaires pour charger le code binaire de RENVERSE ou de NORMAL[1,3];

La taille mémoire nécessaire pour utiliser NORMAL ou RENVERSE dénotée par L, est pour une valeur donnée de n, v, c, p, a et t:

$$L = 7v + n + 7c + 5p + p^2 + a + t \qquad (3.40)$$

Il est clair d'après (3.40) que les paramètres importants sont v, p et a; en particulier, pour des problèmes symétriques où n = 100, on obtient v = 4951 et 7v = 34657, c'est-à-dire presque le tiers de la mémoire totale de le CYBER 173[4]. Les paramètres t, p et a doivent être fixés arbitrairement avant la résolution d'un problème test. Comme rien ne nous permet de prédire a priori les valeurs qu'on doit allouer à ces paramètres pour atteindre la solution optimale, quelques problèmes-tests n'ont pu être résolus à cause d'une valeur trop petite accordée initialement à t, p ou a.

---

(1) Nombre qui demeure fixe au cours de l'optimisation.

(2) Nombre maximal sur une quantité qui varie au cours de l'optimisation.

(3) Même nombre, à peu près égal à 11600.

(4) Qui est approximativement 98,304 mots.

Des travaux récents de Land [69] sur le PVC suggèrent quelques moyens permettant d'économiser de l'espace-mémoire en relaxant certaines variables par l'analyse de leurs coûts réduits en plus de relaxer les contraintes sur les sous-tours pour le PVC d'une façon similaire à la nôtre pour le m-PVC. Mentionnons que nous avons effectué une certaine optimisation de l'utilisation de la mémoire pour nos algorithmes en n'utilisant qu'un seul mot-mémoire pour i, j et c_{ij}. Une amélioration importante consisterait en une spécialisation du code de Land et Powell pour des problèmes avec variables binaires.

Tests avec m fixé

Dans le tableau 3.3 on retrouve les résultats de tests effectués avec l'algorithme RENVERSE lorsque m est fixé à une valeur entre 2 et 10 pour n = 20,30,40. A des fins de comparaison, on a ajouté au bas de ce tableau une ligne donnant les temps de résolution pour les m_ℓ-PVC correspondants.

La première remarque importante qui se dégage de ce tableau est que, avec notre algorithme et pour les valeurs de n testées, le $\bar{m}$-PVC est d'autant plus facile à résoudre que $\bar{m}$ est grand pour une même valeur de n, surtout dans les cas symétrique euclidien et asymétrique. Il semble que lorsque m est fixé à une grande valeur les contraintes de degré au noeud-dépôt sont tellement "fortes" que peu de sous-tours illégaux sont rencontrés lors de l'optimisation. Ceci est probablement dû au fait que dans le graphe de toute solution réalisable d'un $\bar{m}$-PVC il y a $\bar{m}$+n-1 arcs ou arêtes dont $2\bar{m}$ sont incidentes au dépôt. Donc plus $\bar{m}$ est grand, plus la proportion des arcs ou arêtes non incidentes au dépôt est petite, d'où moins de sous-tours illégaux peuvent se former. Mentionnons qu'Ali et Kennington confirment ce fait avec leur méthode [1].

La deuxième remarque qui se dégage du tableau 3.4 est qu'avec notre algorithme RENVERSE le m_ℓ-PVC est plus difficile à résoudre que le $\bar{m}$-PVC, surtout si $\bar{m}$ est grand. Ce phénomène semble d'ailleurs

TABLEAU 3.3

Tests de performance

sur le m̄-PVC

m̄ \ n	20			30			40		
	S-NE(1)	S-E(2)	A(3)	S-NE	S-E	A	S-NE	S-E	A
2	1.30(4)	1.71(4)	3.81(4)	4.71(4)	6.97(4)	11.78(4)	12.01(4)	31.23(4)	33.18(5)
3	1.28	1.50	3.49	5.31	5.87	11.57	11.95	23.76	45.03
4	1.26	1.44	4.17	4.19	5.91	11.63	12.19	23.97	29.91
5	1.27	1.39	3.61	4.54	5.79	12.07	11.61	24.39	27.79
6	1.27	1.23	3.14	4.67	5.76	12.37	11.62	17.23	27.00
7	1.23	1.15	3.25	3.98	4.76	11.46	14.41	15.29	24.95
8	1.28	1.07	3.53	4.70	4.05	10.94	11.63	16.50	26.99
9	1.14	0.97	9.25*	3.90	4.05	11.74	12.07	17.40	27.83
10	1.11	0.85	2.98	4.10	3.82	10.67	11.93	13.54	25.44
m libre	1.2	1.2	3.6	5.2	7.2	17.8	13.2	26.8	33.6

(1) symétrique non-euclidien
(2) symétrique euclidien
(3) asymétrique
(4) temps moyen en secondes pour 5 problèmes
(5) temps moyen en secondes pour 3 problèmes
(*) un des problèmes était exceptionnellement difficile

s'accentuer si n croît. A notre avis, l'explication de ce phénomène est la suivante: nous avons déjà constaté que la valeur optimale de m dans un m_ℓ-PVC est presque[1] toujours 1 et ceci quel que soit le type de problème considéré. Les temps de résolution d'un m_ℓ-PVC devraient donc se rapprocher de ceux du $\bar{m}$-PVC où $\bar{m}$ est petit (disons $\bar{m}$ = 1,2 ou 3) et la remarque précédente nous indique que ce sont ceux-ci qui sont les plus difficiles à résoudre.

Résolution d'un $\bar{m}$-PVC à l'aide de la transformation d'Eilon et al. [35]

On a vu qu'on peut résoudre un m-PVC comme un PVC par l'utilisation de transformations (section 3.2). Le premier désavantage de celles-ci est d'augmenter la taille du problème à traiter. De plus Gavish [44], dans une communication privée, nous a mentionné que les PVC ainsi obtenus sont particulièrement difficiles à résoudre. Le tableau 3.4 suivant ne constitue pas à notre avis une infirmation ni une confirmation de cet énoncé. Nous avons voulu seulement comparer les temps de résolution d'un $\bar{m}$-PVC par 2 méthodes différentes: notre algorithme RENVERSE (méthode B dans le tableau 3.4) résolvant le $\bar{m}$-PVC directement et l'algorithme RENVERSE de Miliotis [92] appliqué au PVC provenant de la transformation du $\bar{m}$-PVC en un PVC par la transformation d'Eilon et al. (méthode A dans 3.5). Le fait que la résolution directe du $\bar{m}$-PVC s'avère beaucoup plus rapide sur des problèms symétriques euclidiens et non-euclidiens avec n = 20,30 ne peut s'avérer une preuve de la pathologie des PVC provenant de transformations car ce phénomène peut être dû aux méthodes employées. Les temps sont donnés en secondes. La valeur de $\bar{m}$ est plus grande ou égale à 2 car si $\bar{m}$ = 1 les deux méthodes sont équivalentes. On remarquera que, dans le tableau 3.5, les temps de résolution de la méthode A augmentent lorsque $\bar{m}$ augmente. Ceci est probablement dû au fait que la méthode A est appliquée sur des problèmes de plus en plus grande taille si $\bar{m}$ augmente[2]. Le phénomène inverse a déjà été constaté pour la méthode B.

---

(1) Toujours dans le cas symétrique euclidien.

(2) Il se peut aussi que la pathologie soulevée par Gavish en cosntitue la cause.

TABLEAU 3.4

Résolution d'un $\bar{m}$-PVC par transformation en un PVC(A) et résolution directe d'un $\bar{m}$-PVC par RENVERSE(B)

$\bar{m}$ \ n	20-S-NE(1)		20-S-E(2)		30-S-NE		30-S-E	
	A	B	A	B	A	B	A	B
2	1.57(3)	1.30(3)	2.05(3)	1.71(3)	5.74(3)	4.71(3)	5.71(4)	5.09(4)
3	2.02	1.28	2.47	1.50	7.76	5.31	8.07	6.09
4	2.56	1.25	2.95	1.44	7.89	4.19	10.44	5.66
5	3.30	1.26	3.37	1.39	9.18	4.54	9.72	5.56
6	3.37	1.26	4.06	1.23	9.93	4.57	12.21	5.97
7	3.93	1.22	4.79	1.15	11.50	3.98	11.90	4.15
8	4.64	1.28	5.33	1.07	14.01	4.70	12.89	4.04
9	5.46	1.14	6.44	0.97	15.56	3.90	13.23	4.46
10	6.17	1.10	6.88	0.85	16.71	4.10	14.54	3.90

3.7 Comparaison de la performance de nos deux algorithmes versus celle des autres méthodes résolvant le m-PVC

Nous avons relevé dans la littérature trois algorithmes optimaux pour le m-PVC. Il s'agit de celui de Svestka et Huckfeldt [119], de celui d'Ali et Kennington [1] et de celui de Gavish et Srikanth [47]. Ces trois algorithmes, décrits brièvement en 3.5, traitent tous du $\bar{m}$-PVC. Nos algorithmes NORMAL et RENVERSE, bien que résolvant le $\bar{m}$-PVC et le m_ℓ-PVC, ont surtout été testés sur le m_ℓ-PVC. Nous sommes d'ailleurs, à notre connaissance, les seuls à avoir développé des méthodes exactes explicitement pour le m_ℓ-PVC.

---

(1) Symétrique non-euclidien.

(2) Symétrique euclidien.

(3) Temps moyen pour 5 problèmes aléatoires.

(4) Temps pour 1 problème aléatoire.

Toute comparaison entre nos algorithmes et ceux des auteurs précédemment cités est difficile à effectuer étant donné que la majorité de nos tests concernent un problème différent et que d'après nos résultats, le $\bar{m}$-PVC est un problème plus facile à résoudre que le m_ℓ-PVC, d'autant plus que $\bar{m}$ est grand relativement à n. L'explication de ce phénomène a d'ailleurs été donnée auparavant dans la section 3.6.5.

Compte tenu de la différence des problèmes traités, de la différence des ordinateurs utilisés et à la lumière de nos résultats et de ceux obtenus par les auteurs précédents, la relaxation que nous avons utilisée pour le m-PVC conduit, à notre avis, à des algorithmes (surtout RENVERSE) qui sont:

1. supérieurs à celui de Gavish et Srikanth sur des m-PVC symétriques euclidiens;
2. inférieurs à celui de Gavish et Srikanth sur des m-PVC symétriques non-euclidiens;
3. nettement supérieurs à ceux de Svestka et Huckfeldt et de Ali et Kennington pour un type quelconque de matrice de coûts de transport.

On a vu en 3.2 qu'on peut toujours, quel que soit le cas particulier[1] considéré, transformer un m-PVC en un PVC. La question qu'il faut alors se poser est la suivante. Vaut-il mieux aborder directement le m-PVC ou bien le transformer tout d'abord en un PVC pour ensuite le résoudre comme tel? Selon Ali et Kennington [1] la réponse à cette question est encore inconnue. A notre avis les éléments de réponse suivants peuvent être apportés sur cette question:

1. en considérant les algorithmes les plus performants à ce jour pour le PVC asymétrique et pour le m-PVC de même type, ce dernier problème devrait actuellement être abordé par transformation;

---

(1) $\bar{m}$-PVC, m_ℓ-PVC, coûts fixes nuls ou non-nuls, etc.

2. en considérant nos résultats sur le m_ℓ-PVC et le $\bar{m}$-PVC non-euclidiens et ceux de Gavish et Srikanth [47] sur le $\bar{m}$-PVC non-euclidien, le m-PVC non-euclidien devrait présentement être abordé directement;

3. en considérant nos résultats sur le $\bar{m}$-PVC euclidien, ce dernier problème devrait être résolu directement. Notons que le m_ℓ-PVC euclidien est équivalent au PVC.

Le point 1 précédent tient compte du fait qu'il existe présentement des algorithmes [2,15] ayant résolu certains PVC asymétriques de quelque 325 villes. A notre connaissance les m-PVC asymétriques de plus grande taille résolus contiennent 100 villes [1].

Les points 2 et 3, concernant les m-PVC symétriques, découlent des constatations suivantes. Si des approches comme la nôtre et celle de Miliotis [90], basées toutes deux sur la programmation linéaire en nombres entiers, sont utilisées, il est clair que le m_ℓ-PVC doit être abordé directement car d'une part ce problème est une relaxation du PVC et d'autre part toute transformation implique une augmentation du nombre de villes. Puisque, avec ce type d'approche, nos résultats démontrent que le m_ℓ-PVC est plus difficile à résoudre que le $\bar{m}$-PVC, ce dernier problème doit aussi être abordé directement si une approche comme la nôtre est employée. Notons que les résultats de Gavish et Srikanth [47], avec une approche différente de la nôtre basée sur la notion d'un m-arbre, corroborent cette dernière affirmation pour le $\bar{m}$-PVC non-euclidien.

3.8 Conclusion

A partir du principe général que nous avons mis au point, nous avons construit deux algorithmes traitant du m-PVC. Ces deux algorithmes (surtout la version RENVERSE) démontrent clairement que la relaxation que nous avons choisie pour le m-PVC est très efficace et ils sont globalement compétitifs avec les autres méthodes existantes résolvant le m-PVC. Le principal avantage de ce type d'approche est sa flexibilité. Les

algorithmes qui en découlent pour le m-PVC sont beaucoup plus flexibles que tous les autres construits pour ce problème. Ils s'adaptent facilement à un type quelconque de matrice des distances, ils peuvent traiter avec des modifications mineures le $\bar{m}$-PVC, le m_ℓ-PVC sans coûts fixes, le m_ℓ-PVC avec coûts fixes tous égaux ou possiblement distincts et des bornes inférieures et/ou supérieures peuvent être facilement imposées sur m. De plus des méthodes similaires s'avèrent très utiles dans la résolution de plusieurs autres extensions du PVC tel qu'en témoignent les chapitres 4, 5 et 6 de ce texte. Remarquons de plus que des algorithmes pour le m-PVC basés sur la notion d'un m-arbre et la relaxation lagrangienne, tels [1,47], s'adaptent difficilement au m_ℓ-PVC non-euclidien. Si le nombre de commis est borné par $\bar{m}$, la résolution du lagrangien devrait comporter, à notre avis, $\bar{m}$ recherches d'un m-arbre de poids minimal, m variant de 1 jusqu'à $\bar{m}$, au lieu d'un seul calcul de ce type pour le $\bar{m}$-PVC; ce qui réduirait de beaucoup la performance de telles méthodes.

Toutes les extensions du PVC traitées dans ce texte sont résolues de façon similaire au m-PVC. Dans ce type d'approche la méthode du simplex est l'outil de base résolvant les différentes relaxations obtenues. Une telle procédure sera toujours plus lente dans le cas asymétrique que dans le cas symétrique car le nombre de variables et de contraintes est approximativement doublé. De plus, dans le cas asymétrique, les contraintes de degré auront habituellement la structure des contraintes d'un problème de flot, et ceci même si leur forme varie d'une extension à l'autre. Il serait, à notre avis, avantageux de tenir compte de ce fait dans le développement d'algorithmes pour de telles extensions. Pour cette raison, dans les chapitres suivants, la matrice des coûts de transport sera toujours symétrique dans les tests de performance effectués sur ordinateur. Notons cependant que tous nos travaux s'étendent facilement au cas asymétrique.

CHAPITRE 4

LE PROBLEME DES m VOYAGEURS DE COMMERCE ASSUJETTIS A UNE RESTPICTION DE CAPACITE (c.m-PVC)

4. LE PROBLEME DES m VOYAGEURS DE COMMERCE ASSUJETTIS A UNE RESTRICTION DE CAPACITE (c.m-PVC)

4.1 Introduction

Nous avons considéré aux chapitres 2 et 3 de ce texte le problème du voyageur de commerce (PVC) et celui des m voyageurs de commerce (m-PVC). Ces deux problèmes possèdent de nombreuses applications pratiques dans les problèmes de transport où les routes d'une flotte de véhicules devant visiter certains clients doivent être optimisées. Le PVC et le m-PVC ne peuvent toutefois pas tenir compte de plusieurs particularités rencontrées fréquemment dans ce type de problèmes surtout au niveau des caractéristiques de la flotte de véhicules et des contraintes à respecter, notamment lorsque:

1. les véhicules sont différents quant à leurs coûts de transport inter-villes et leurs temps de transport inter-villes; des données du type c_{ij}^k et t_{ij}^k sont alors connues où c_{ij}^k (t_{ij}^k) est le coût (temps) d'effectuer le trajet de la ville i vers la ville j par le véhicule k; quelquefois les c_{ij}^k et les t_{ij}^k sont assimilés l'un à l'autre;

2. la capacité de livraison des véhicules est possiblement différente et restreint le nombre de villes qu'ils peuvent visiter;

3. un temps maximum ou un coût maximum est imposé sur les routes des véhicules, i.e. la somme des t_{ij}^k (c_{ij}^k) subie sur une route est alors bornée supérieurement par une constante;

4. les routes des véhicules peuvent débuter à plusieurs dépôts différents[1] et/ou se terminer à un endroit distinct de leur point de départ;

5. certaines villes doivent être visitées avant d'autres;

---

(1) Le cas de plusieurs dépôts sans restriction de capacité sera traité au chapitre 5 de ce texte.

6. la visite à chaque ville doit se produire dans un intervalle de temps donné (contraintes de fenêtres).

Les quatre premières caractéristiques précédentes sont inhérentes aux problèmes connus sous l'appellation anglaise "vehicle routing problems". Lorsqu'on adjoint à ces dernières les caractéristiques 5 et 6, on dira qu'il s'agit de "vehicle routing and scheduling problems". Nous avons traduit cette dernière terminologie par les problèmes d'expédition d'une flotte de véhicules (PEV). On retrouve dans les articles [11,12,20,52] une liste exhaustive des différentes propriétés et applications des PEV.

Les commentaires précédents expliquent pourquoi la résolution exacte du PVC ou du m-PVC peut produire dans la réalité une solution approximative. L'utilisation de modèles plus complexes où un sous-ensemble des caractéristiques 1 jusqu'à 6 est inclus conduit toutefois à des problèmes beaucoup plus difficiles à résoudre optimalement. D'ailleurs, les algorithmes rencontrés dans la littérature résolvant de tels problèmes sont en majorité des heuristiques.

Nous étudions dans ce chapitre la construction d'un modèle et son utilisation dans un algorithme optimal pour un des cas particuliers les plus importants des PEV. Il s'agit du c.m-PVC où la caractéristique 2 précédente est adjointe au modèle du m-PVC avec des véhicules de capacité identique. Le c.m-PVC est défini en 4.2.

4.2 Définition du c.m-PVC

Le c.m-PVC est une extension directe du m-PVC traité au chapitre 3. Soit G = (N,A) le graphe d'ordre n associé au m-PVC et au c.m-PVC. Dans le m-PVC, m sous-tours[1] de G = (N,A) doivent être construits de façon à respecter les contraintes suivantes:

---

(1) Ces m sous-tours forment ce que nous avons appelé au chapitre 3 un m-tour.

1. chaque ville autre que le dépôt n'est contenue que dans un seul sous-tour;

2. chacun des sous-tours doit contenir le dépôt.

Dans le c.m-PVC, l'ensemble des villes faisant partie d'un sous-tour donné est assujetti à une contrainte supplémentaire. Il s'agit d'une contrainte dite de "capacité" que l'on peut définir ainsi. Soient $d_1,\ldots d_n$ des entiers non-négatifs représentant respectivement la demande pour un certain bien des villes dans $N = \{1,2,\ldots,n\}$ où la ville 1 sert de dépôt. Soit D la capacité de livraison, pour le bien considéré, des véhicules destinés à parcourir les itinéraires indiqués par les m sous-tours. Dans le c.m-PVC, la restriction supplémentaire suivante est imposée sur la construction des m sous-tours:

3. la demande totale des villes d'un sous-tour donné doit être inférieure ou égale à D.

On suppose ici que $d_1 = 0$[1], que $0 < d_2,\ldots,d_n \leq D$ et que m, le nombre de sous-tours (ou véhicules), n'est pas fixé a priori. On verra qu'il serait facile de considérer m fixé dans notre modèle ainsi que dans notre algorithme. Nous avons choisi cependant d'étudier un problème plus général. De façon à se conformer à la terminologie usuelle concernant le c.m-PVC, on dira que chaque sous-tour contenant le dépôt constitue le "tour" d'un commis ou la route d'un commis.

L'objectif à minimiser dans le c.m-PVC est la somme des coûts de transport inter-villes induits par les m tours. La matrice $C = (c_{ij})$ contiendra les coûts de transport entre chaque paire possible de villes. On pourrait aisément tenir compte dans l'objectif du nombre de commis utilisés en y adjoignant un terme de la forme gm où g serait le coût fixe d'utilisation d'un commis. Ceci n'ajoute aucune difficulté supplémentaire dans la formulation et la résolution du c.m-PVC.

---

(1) On suppose que le dépôt n'est qu'un lieu de départ et d'arrivée pour les véhicules et que sa demande est nulle ($d_1 = 0$).

Parmi les premiers chercheurs à traiter du c.m-PVC, on retrouve Dantzig et Ramser [29] qui ont donné à ce problème l'appellation anglaise "The truck dispatching problem". On rencontre aussi quelquefois dans la littérature les appellations "the vehicle scheduling problem" [14], "the vehicle dispatching problem" [18] ou même "the delivery problem" [57]. Notons que ces trois dernières appellations font quelquefois référence à un problème un peu différent du c.m-PVC où m est fixé et/ou les véhicules peuvent posséder des capacités de livraison différentes.

Les applications du c.m-PVC sont nombreuses malgré sa simplicité par rapport à d'autres modèles du type PEV. Mentionnons entre autres la cueillette des lettres et des colis dans les boîtes postales ainsi que la construction d'itinéraires pour les autobus scolaires. Remarquons que, dans cette dernière application, il ne s'agit pas de livrer des biens aux différentes "villes" mais bien de les recueillir. En autant qu'il s'agisse uniquement de livraisons ou uniquement de cueillettes, cette distinction ne modifie cependant en rien la nature du c.m-PVC.

Notons en dernier lieu que le c.m-PVC est équivalent au m-PVC si $D > \sum_{i=2}^{n} d_i$ et aussi au PVC si en plus m est fixé à la valeur 1. On verra toutefois qu'en général le c.m-PVC est très différent de ces deux derniers problèmes. Les principales causes de ce phénomène seront explicitées dans les prochaines sections.

4.3 Formulations mathématiques pour le c.m-PVC

4.3.1 Le modèle de Balinski et Quandt [4]

La plupart des modèles mathématiques mis au point pour le c.m-PVC peuvent être classifiés selon qu'ils traitent du cas où C est asymétrique ou bien du cas où C est symétrique. Il existe toutefois un modèle dû à Balinski et Quandt [4] qui est indépendant de cette caractéristique de C. Ce modèle présuppose que tous les tours réalisables peuvent être énumérés. Ainsi, pour le c.m-PVC, il s'agit d'un tour dont la demande totale des villes qui en font partie est plus petite ou égale à la

capacité de livraison d'un véhicule. Pour décrire le modèle de Balinski et Quandt la notation suivante est nécessaire:

T le nombre total de tours réalisables

$$x_j = \begin{cases} 1 \text{ si le } j^e \text{ tour est choisi } (j = 1,\ldots,T) \\ 0 \text{ sinon} \end{cases}$$

$$\delta_{ij} = \begin{cases} 1 \text{ si la } i^e \text{ ville fait partie du } j^e \text{ tour}^{(1)} & (i = 2,\ldots,n;\ j = 1,\ldots,T) \\ 0 \text{ sinon} \end{cases}$$

c_j le coût du j^e tour

En pratique, les tours réalisables sont engendrés de la façon suivante: pour chaque ensemble $S_j \subseteq N$ tel que $\sum_{i \in S_j} d_i \leq D$, un PVC avec $N = S_j$ est résolu. Le tour réalisable associé à S_j ainsi que son coût (c_j) sont donnés par la solution optimale de ce PVC et par son coût optimal.

Le modèle de Balinski et Quandt peut s'écrire:

$$(P1) \quad \begin{cases} \text{Min } Z = \sum_{j=1}^{T} c_j x_j & (4.1) \\ \text{assujettie à:} \\ \sum_{j=1}^{T} \delta_{ij} x_j = 1^{(2)} \quad (i = 2,\ldots,n) & (4.2) \\ x_j = 0,1 \quad (j = 1,\ldots,T) & (4.3) \end{cases}$$

Le modèle P1 s'explique facilement. Les variables x_j étant associées à des tours réalisables, il n'est pas nécessaire d'y inclure des contraintes assurant que ceux qui sont choisis sont légaux. Il suffit

---

(1) $\delta_{1j} = 1$ pour tout j car le dépôt doit être contenu dans tout tour. Les δ_{ij} sont connus une fois les tours énumérés.

(2) On peut permettre la possibilité qu'une ville fasse partie de plus d'un tour en mettant "$\geq$" au lieu de "$=$". Il s'agit alors d'un problème de recouvrement d'ensembles [108] ("set covering").

uniquement de s'assurer que chacune des villes est contenue dans exactement un tour. (4.1) jusqu'à (4.3) définissent un problème de "set partitioning" et P1 peut être résolu, en théorie, par tout algorithme de résolution de ce modèle [3]. L'avantage de ce modèle réside dans sa flexibilité à pouvoir tenir compte de contraintes supplémentaires sur les tours[1]. La forme de P1 demeure inchangée mais la génération des tours réalisables est plus difficile. En outre, plus la réalisibilité des tours est restreinte par des contraintes, plus T sera "petit" et moins grand sera la nombre de variables dans P1. Malgré cet avantage théorique, P1 est peu utile d'un point de vue algorithmique car, comme le mentionne Christofides [16], T demeure en pratique extrêmement grand même si, en plus d'une contrainte de capacité, des contraintes de temps maximum, des contraintes de fenêtre, etc. sont imposées sur la construction des tours.

4.3.2 Notre modèle pour le c.m-PVC symétrique

A notre connaissance, le seul modèle mathématique formulant directement le c.m-PVC symétrique en n'utilisant que des variables x_{ij} où $i < j$ est celui que nous avons mis au point. Il s'écrit:

(P2)

$$\text{Min } Z = \sum_{i<i} c_{ij}\, x_{ij} \tag{4.4}$$

assujettie à:

$$\sum_{j=2}^{n} x_{1j} = 2m \tag{4.5}$$

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2 \qquad (k = 2,\ldots,n) \tag{4.6}$$

$$\sum_{\substack{i\in S,\, j\in S \\ i<j}} x_{ij} \leq |S| - \left\lceil \frac{\sum_{i\in S} d_i}{D} \right\rceil \qquad (S \subset N-\{1\}) \tag{4.7}$$

$$x_{1j} = 0,1,2 \qquad (j = 2,\ldots,n) \tag{4.8}$$

---

(1) Comme les caractéristiques 1, 3, 4, 5 et 6 de la section 4.1.

$$\begin{cases} x_{ij} = 0,1 & (2 \leq i<j \leq n) \quad (4.9) \\ m \geq 1 \ (1) & \quad (4.10) \end{cases}$$

Théorème 4.1

Le modèle P2 est valide pour le c.m-PVC symétrique.

Preuve

Les équations (4.4), (4.5), (4.6) et (4.8) jusqu'à (4.10) sont identiques aux contraintes de degré et aux contraintes de définition sur les variables du modèle pour le m-PVC symétrique que nous avons construit. Elles s'expliquent donc de façon similaire et le lecteur est référé au chapitre 3, section 3.4.2, pour plus de détails. La preuve de la validité de (4.7) peut s'effectuer de la façon suivante. Soit $\min_S = \left\lceil \frac{\sum_{i \in S} d_i}{D} \right\rceil$

Soit $S \subset N - \{1\}$. Puisque le dépôt fait partie de $\bar{S}$, il y aura forcément au minimum $\min_S$ véhicules qui effectueront chacun un aller de $\bar{S}$ à S et un retour de S à $\bar{S}$, ce qui nous permet d'écrire que:

$$\sum_{\substack{i \in S, j \in \bar{S} \\ \text{ou} \\ j \in S, i \in \bar{S} \\ i<j}} x_{ij} \geq 2\min_S \quad (4.11)$$

Puisque tous les sommets de S ont un degré de 2 si (4.6) est satisfaite, le théorème 2.2 nous permet d'écrire directement:

$$\sum_{\substack{i \in S, j \in \bar{S} \\ \text{ou} \\ i \in S, i \in \bar{S} \\ i<j}} x_{ij} \geq 2\min_S - \sum_{\substack{i \in S, j \in S \\ i<j}} x_{ij} \leq |S| - \min_S.$$

■

Les contraintes (4.7) et (4.11) généralisent respectivement (2.24) et

(1) m prendra une valeur entière par (4.8) et (4.5).

(2.26), dues à Dantzig et al. [28], au cas du c.m-PVC. Elles permettent d'éliminer tout sous-tour illégal qui peut survenir si (4.7) est omise de P2. Il y a deux types de sous-tours illégaux dans le c.m-PVC:

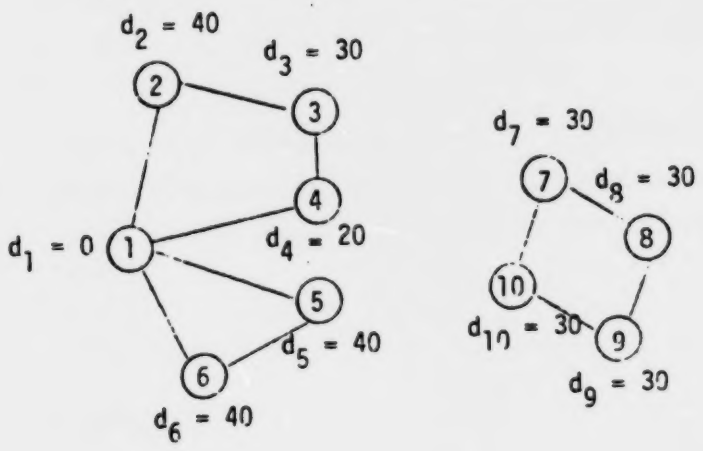
Type 1: sous-tour ne contenant pas le dépôt;

Type 2: sous-tour contenant le dépôt (c'est-à-dire un tour) mais dont la demande[1] dépasse D.

A titre d'exemple, considérons une solution irréalisable pour le c.m-PVC illustrée dans la figure 4.1 où (4.5), (4.6), (4.8), (4.9), (4.10) sont satisfaites et où m = 2, D = 80, n = 10.

FIGURE 4.1

Exemple d'une solution irréalisable pour le c.m-PVC



Soient $S_1 = \{2,3,4\}$ et $S_2 = \{7,8,9,10\}$

On a:
$$\begin{cases} \min_{S_1} = 2,\ \min_{S_2} = 2,\ |S_1| = 3,\ |S_2| = 4 \\ \sum_{i \in S_1, j \in S_1, i<j} x_{ij} = 2 > |S_1| - \min_{S_1} = 3-2 = 1 \\ \sum_{i \in S_2, i \in S_2, i<j} x_{ij} = 4 > |S_2| - \min_{S_2} = 4-2 = 2 \end{cases}$$

(1) La demande totale des villes qui en font partie.

Les 2 contraintes de (4.7) basées sur S_1 et sur S_2 sont donc violées par la solution illustrée dans la figure 4.1. Notons qu'un sous-tour illégal du type 1 avec S comme ensemble de villes peut être éliminé aussi par une contrainte de Dantzig et al. du type:

$$\sum_{i\in S, j\in S \atop i<j} x_{ij} \leq |S|-1 \qquad (4.12)$$

Toutefois, si $\min_S > 1$, (4.7) est plus forte que (4.12). Ces commentaires s'appliquent aussi à S_2.

Notons en dernier lieu que si $D > \sum_{i=2}^{n} d_i$ on a alors $\min_S = 1$ pour tout $S \subset N-\{1\}$ et le modèle P2 précédent se réduit à notre modèle donné en 3.4.2, pour le m-PVC symétrique.

4.3.3 Cas asymétrique

Les principales formulations mathématiques du c.m-PVC asymétrique seront décrites dans cette section sauf celle de Gavish et Graves [45] et la nôtre qui seront étudiées de façon plus détaillée dans la section 4.4.

Un des modèles les plus fréquemment utilisés par les chercheurs pour le c.m-PVC asymétrique est dû à Golden et al. [52]. Bien que dans le c.m-PVC les véhicules soient identiques du point de vue de leur capacité et de leurs coûts de transport, ce modèle utilise des variables binaires x_{ij}^k pour indiquer le passage du k^e véhicule de la ville i à la ville j[1]. Le nombre de véhicules est supposé borné supérieurement et nous dénoterons cette borne par $\bar{m}$. Le modèle de Golden et al. peut s'écrire de la façon suivante. Le rôle de chaque contrainte sera indiqué immédiatement avant son écriture mathématique.

$$\left[\text{Min } Z = \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{\bar{m}} c_{ij}\, x_{ij}^k \right. \qquad (4.13)$$

---

(1) Encore ici on suppose $i \neq j$.

assujettie à:

(4.14): un et un seul véhicule entre dans une ville autre que le dépôt:

$$\sum_{i=1}^{n} \sum_{k=1}^{\bar{m}} x_{ij}^{k} = 1 \qquad (j = 2,\ldots,n)$$

(4.15): un seul et un seul véhicule sort d'une ville autre que le dépôt:

$$\sum_{j=1}^{n} \sum_{k=1}^{\bar{m}} x_{ij}^{k} = 1 \qquad (i = 2,\ldots,n)$$

(4.16): si un véhicule entre dans une ville, il doit en sortir:

$$\sum_{i=1}^{n} x_{ip}^{k} - \sum_{j=1}^{n} x_{pj}^{k} = 0 \qquad (k = 1,\ldots,\bar{m};\ p = 1,\ldots,n)$$

(P3)

(4.17): la capacité de chaque véhicule est respectée:

$$\sum_{i=2}^{n} d_i \left(\sum_{j=1}^{n} x_{ij}^{k} \right) \leq D \qquad (k = 1,\ldots,\bar{m})$$

(4.18): chacun des véhicules peut sortir du dépôt au maximum une fois:

$$\sum_{j=2}^{n} x_{1j}^{k} \leq 1 \qquad (k = 1,\ldots,\bar{m})$$

(4.19): chacun des véhicules peut entrer au dépôt au maximum une fois:

$$\sum_{i=2}^{n} x_{i1}^{k} \leq 1 \qquad (k = 1,\ldots,\bar{m})$$

(4.20): aucun sous-tour illégal du type 1 ne doit se former:

soit $x_{ij} = \sum_{k=1}^{\bar{m}} x_{ij}^{k}$; les x_{ij} peuvent être assujetties dans

(4.20) aux contraintes de Dantzig et al. (2.7) ou aux contraintes de Miller et al. (2.13) qui empêchent ce type de sous-tour de se former. Les sous-tours illégaux de type 2 sont prévenus par (4.17).

(4.21): $x_{ij}^{k} = 0,1 \qquad (i,j = 1,\ldots,n;\ k = 1,\ldots,\bar{m})$

Les contraintes (4.15) et (4.19) sont clairement redondantes[1]. Les variables x_{ij}^{k}, en identifiant lequel des $\bar{m}$ véhicules circule sur un arc donné, facilitent l'écriture des contraintes de capacité et permettent même, avec des modifications mineures, l'utilisation d'une flotte de véhicules avec des capacités non identiques [52]. De plus les contraintes de (4.17) sont en très petit nombre, notamment $\bar{m}$. On verra en (4.4) le modèle que nous avons établi pour le c.m-PVC asymétrique. Ce modèle, dénoté P4, est similaire à P2 traitant du cas symétrique et contient un nombre exponentiel en terme de n de contraintes assurant que la capacité des véhicules est respectée. Si dans P3 les contraintes de Dantzig et al. sont utilisées dans (4.20), P3 et P4 contiennent un nombre de contraintes du même ordre de grandeur. Si par contre les contraintes de Miller et al. servent dans (4.20), P3 contiendra un nombre polynomial de contraintes en termes de n et $\bar{m}$. Toutefois dans ce cas, malgré un moins grand nombre de contraintes que dans P4, le nombre de variables et de contraintes dans P3, même pour n et $\bar{m}$ petits, le rend extrêmement difficile à résoudre directement. D'ailleurs, à notre connaissance, aucun chercheur n'a utilisé P3 pour l'élaboration d'un algorithme optimal pour le c.m-PVC. On verra que des modèles comme P2 et P4 s'avèrent beaucoup plus utiles pour la construction de tels algorithmes.

Nous avons relevé dans la littérature deux autres modèles pour le c.m-PVC asymétrique. Le premier est dû à Fisher et Jaikumar [36] et est

---

(1) (4.15) est impliquée par (4.14) et (4.16) tandis que (4.19) est induite par (4.16) et (4.18).

très similaire à celui de Golden et al. Fisher et Jaikumar utilisent, en plus des variables x_{ij}^k, des variables binaires y_{ik} identifiant le véhicule particulier k qui visite la ville i. Evidemment de telles variables ne sont pas nécessaires car le véhicule visitant une ville donnée i est aussi identifié par la variable x_{ij}^k ayant une valeur 1 dans la solution optimale. Toutefois, cet artifice de formulation permet à Fisher et Jaikumar d'en arriver à un problème d'affectation généralisée [105] lorsque (4,20) est relaxée et de construire un heuristique efficace pour le c.m-PVC.

Le dernier modèle dont nous voulons faire rapidement mention ici a été élaboré par Garvin et al. [41]. Il se différencie nettement des autres modèles pour le c.m-PVC par son choix de variables de décision. Leur modèle utilise des variables y_{ijk} continues représentant la quantité de biens transportés de la ville i à la ville j et destinés à la ville k en plus des habituelles variables binaires x_{ij}. Le modèle de Garvin et al. contient, à notre avis, un nombre inutilement grand de variables et n'a pas, à notre connaissance, servi à la mise sur pied d'algorithmes optimaux pour le c.m-PVC. Nous n'en ferons donc plus mention dans la suite du texte.

4.4 Notre modèle et celui de Gavish et Graves [45] pour le c.m-PVC asymétrique

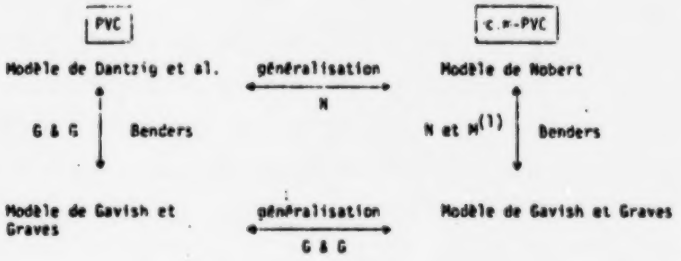
4.4.1 Introduction

Le matériel présenté dans cette section se divise en 2 parties:

1. description de notre modèle et de celui de Gavish et Graves pour le c.m-PVC asymétrique en 4.4.2;
2. étude de la relation entre ces deux modèles en 4.4.3.

On verra en 4.4.2 que le modèle de Gavish et Graves se différencie du nôtre dans la forme des contraintes utilisées pour éliminer les sous-tours illégaux de type 1 ou de type 2. On démontrera toutefois en 4.4.3 que ces deux modèles sont étroitement reliés entre eux. De façon

plus précise, il sera prouvé qu'en appliquant la technique de décomposition de Benders [9] au modèle de Gavish et Graves on retrouve les contraintes d'élimination de sous-tours illégaux que nous avons mises au point pour c.m-PVC asymétrique. Ce théorème nous permet d'établir le schéma suivant illustrant les relations connues entre quelques modèles pour le PVC et le c.m-PVC:



Le lien $\xleftarrow[G\&G]{Benders}$ signifie que le modèle de Dantzig et al. pour le PVC se déduit du modèle de Gavish et Graves pour ce même problème par la technique de décomposition de Benders et que ce résultat a été prouvé par Gavish et Graves (G&G). De même le lien $\xleftarrow[N]{généralisation}$ indique que Nobert (N) a généralisé le modèle de Dantzig et al. pour le PVC au cas du c.m-PVC. Le reste du schéma s'interprète de façon similaire.

1.4.2 Description de notre modèle et de celui de Gavish et Graves pour le c.m-PVC asymétrique

Notre modèle généralise les contraintes de Dantzig et al. pour le PVC asymétrique au cas du c.m-PVC asymétrique tandis que celui de Gavish et Graves généralise le modèle de ces mêmes auteurs pour le PVC asymétrique au cas du c.m-PVC asymétrique. Notre modèle et celui de Gavish et Graves possèdent la forme générale suivante:

---

(1) Il existe deux preuves distinctes et indépendantes que notre modèle pour le c.m-PVC se déduit de celui de Gavish et Graves pour ce même problème à l'aide de la technique de décomposition de Benders. Magnanti [87] utilise le lemme de Farkas pour arriver à ce résultat tandis que nous présentons ici une preuve différente.

(IP4)

$$\text{Min } Z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{4.22}$$

<u>assujettie à</u>:

$$\sum_{j=2}^{n} x_{1j} = m \tag{4.23}$$

$$\sum_{i=2}^{n} x_{i1} = m \tag{4.24}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad (j = 2,\ldots,n) \tag{4.25}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad (i = 2,\ldots,n) \tag{4.26}$$

aucun sous-tour illégal de type 1 ou de type 2 ne doit être présent. (4.27)

$$x_{ij} = 0,1 \qquad (i,j = 1,\ldots,n) \tag{4.28}$$

$$m \geq 0 \tag{4.29}$$

Les contraintes que nous utilisons dans (4.27) sont similaires à (4.7) pour le cas symétrique. Elles s'écrivent:

$$\sum_{i\in S, j\in S} x_{ij} \leq |S| - \min_S \qquad (S \subset N-\{1\}) \tag{4.30}$$

où $\min_S$ est défini comme précédemment.

La preuve de la validité de (4.30) est similaire à celle que l'on retrouve dans le théorème 4.1. Le dépôt étant un élément de $\bar{S}$ on a dans le cas asymétrique:

$$\sum_{i\in S, i\in \bar{S}} x_{ij} \geq \min_S \qquad (S \subset N-\{1\}) \tag{4.31}$$

car il y aura forcément au minimum $\min_S$ passages de S vers $\bar{S}$. Par le théorème 2.1 du chapitre 2, il s'ensuit directement que pour tout $S \subset N-\{1\}$:

$$\sum_{i\in S, j\in\bar{S}} x_{ij} \geq \min_S \quad - \quad \sum_{i\in S, j\in S} x_{ij} \leq |S| - \min_S$$

ainsi (4.30) est valide pour le c.m-PVC asymétrique.

Les contraintes (4.30) et (4.31) généralisent respectivement les contraintes (2.9) et (2.7) pour le PVC asymétrique au cas du c.m-PVC du même type. Tout comme pour P2, si $D > \sum_{i=2}^{n} d_i$ alors P4 où (4.30) est utilisé dans (4.27) se réduit au modèle P1 que nous avons mis au point pour le m-PVC asymétrique (section 3.3.3).

Gavish et Graves utilisent dans (4.27) les contraintes suivantes:

$$\sum_{\substack{j=1\\ j\neq i}}^{n} y_{ij} - \sum_{\substack{j=1\\ j\neq i}}^{n} y_{ji} = d_i \qquad (i = 2,\ldots,n) \qquad (4.32)$$

$$0 \leq y_{ij} \leq D x_{ij} \qquad (i \leq i\neq j \leq n) \qquad (4.33)$$

Les variables y_{ij} $(i\neq j)$ seront appelées les variables de flot. Les contraintes (4.32) et (4.33) généralisent respectivement les contraintes de ces mêmes auteurs (2.19) et (2.20) pour le PVC asymétrique au cas du c.m-PVC asymétrique. Le modus operandi de (4.32) et (4.33) peut être expliqué à l'aide des remarques suivantes:

1. (4.32) et (4.33) empêchent la formation de toute route ne contenant pas le dépôt. Pour s'en convaincre, considérons le sous-tour illégal de type 1 $(i_1, i_2, \ldots, i_{k-1}, i_k, i_1)$ où $i_j \neq 1$ pour $j = 1,\ldots,k$ (≥ 2). Soit $S = \{i_1,\ldots,i_k\}$. On a clairement:

$$\begin{cases} x_{i_j,i_{j+1}} = 1 \text{ pour } j = 1, k-1 \\ x_{i_k,i_1} = 1 \\ x_{i_j,i_p} = 0 \text{ pour tout } i_j, i_p \in S \text{ où } \begin{cases} p \neq j+1 \text{ si } i_j \neq i_k \\ p \neq i_1 \text{ si } i_j = i_k \end{cases} \end{cases}$$

On peut donc conclure par (4.33) que:

$$y_{i_j,i_p} = 0 \text{ pour tout } i_j,i_p \in S \text{ où } \begin{cases} p \neq j+1 \text{ si } i_j \neq i_k \\ p \neq i_1 \text{ si } i_j = i_k \end{cases}$$

de plus puisque (4.32) est satisfaite, il faut:

$$\begin{cases} y_{i_1,i_2} - y_{i_k,i_1} = d_{i_1} \\ y_{i_2,i_3} - y_{i_1,i_2} = d_{i_2} \\ \vdots \\ y_{i_k,i_1} - y_{i_{k-1},i_k} = d_{i_k} \end{cases}$$

on obtient donc:

$$0 = \sum_{j=1}^{k} d_{i_j}$$

ce qui est une contradiction.

2. (4.32) et (4.33) empêche la formation de tout sous-tour illégal de type 2. Soit $(1,i_1,i_2,\ldots,i_k,1)$ un tel sous-tour avec $k \geq 1$ et $i_j \neq 1$ pour $j = 1,\ldots,k$. Puisqu'il s'agit d'un sous-tour de type 2, on peut écrire:

$$\sum_{j=1}^{k} d_{i_j} > D$$

. De plus, si (4.32) et (4.33) sont respectées, on obtient de façon similaire à la remarque 1:

$$\begin{cases} y_{i_1,i_2} - y_{1,i_1} = d_{i_1} \\ y_{i_2,i_3} - y_{i_1,i_2} = d_{i_2} \\ \vdots \\ y_{i_k,1} - y_{i_{k-1},i_k} = d_{i_k} \end{cases}$$

En additionnant ces dernières égalités, on déduit:

$$y_{i_k,1} - y_{1,i_1} = \sum_{j=1}^{k} d_{i_j}$$

ce qui implique:

$$y_{i_k,1} = \sum_{j=1}^{k} d_{i_j} + y_{1,i_1} \overset{(1)}{\geq} \sum_{j=1}^{k} d_{i_j} > 0$$

ce qui contredit (4.33).

3. S'il existe une solution réalisable pour le c.m-PVC, il existe un choix de valeurs pour les variables x_{ij} et y_{ij} satisfaisant aux contraintes de P4 où (4.27) est remplacée par (4.32) et (4.33). Les variables x_{ij} prennent leurs valeurs selon leur définition tandis qu'on peut assigner de la façon suivante des valeurs aux variables y_{ij}. Soit $(i_0=1,i_1,\ldots,i_k,1)$ un des tours de cette solution réalisable où $i_j \neq 1$ pour $j = 1,\ldots,k$ et où $k \geq 1$.

Posons:
$$\left\{ \begin{array}{l} d_{i_0} = d_1 = 0,\ i_{k+1} = i_0,\ y_{1,i_1} = 0 \\ y_{i_j,i_{j+1}} = y_{i_{j-1},i_j} + d_{i_j} \text{ pour } j = 1,\ldots,k \end{array} \right\} \quad (4.34)$$

La variable de flot associée à ce tour[2] à laquelle on accorde la plus grande valeur est $y_{i_k,i_0} = \sum_{j=1}^{k} d_{i_j}$. Par hypothèse cette quantité est inférieure ou égale à D. Ainsi toutes les variables de flot touchées par (4.34) ont une valeur satisfaisant à (4.33). En procédant de façon similaire pour chaque route et en posant $y_{ij} = 0$ si $x_{ij} = 0$ on obtient des valeurs pour les variables de flot qui satisfont à (4.32) et à (4.33); en particulier notons que la deuxième ligne de (4.34) se réécrit:

---

(1) Ici on se sert de la non-négativité des variables y_{ij}.

(2) C'est-à-dire la variable y_{ij} où $i,j \in S = \{1,i_1,\ldots,i_k\}$.

$$y_{i_j,i_{j+1}} - y_{i_{j-1},i_j} = d_{i_j} \text{ pour } j = 1,\ldots,k$$

et (4.32) est clairement satisfaite.

Cette dernière remarque complète notre description du modèle de Gavish et Graves. Nous voudrions souligner un résultat intéressant obtenu par Gavish et Shlifer [46] qui s'applique à notre modèle ainsi qu'à celui de Gavish et Graves. Gavish et Shlifer ont prouvé [46] qu'en remplaçant (4.22) par:

$$\text{Max } Z = \sum_{i=1}^{n} \sum_{j=1}^{n} s_{ij} x_{ij} \qquad (4.35)$$

où:

$$s_{ij} = v + c_{i,1} + c_{1,j} - c_{ij} \quad (1 \leq i \neq j \leq n) \qquad (4.36)$$

alors la solution obtenue en résolvant P4 avec cette nouvelle fonction objective est identique à la solution optimale de P4. Les quantités $(s_{ij}-v)$ de (4.36) représentent les "savings" utilisés dans l'heuristique de Clarke et Wright [25] pour le c.m-PVC asymétrique. Cette méthode approximative pour le c.m-PVC sera décrite un peu plus en détails dans la section 4.5.1 de ce chapitre.

4.4.3 Application de la technique de décomposition de Benders [9] au modèle de Gavish et Graves.

Le modèle de Gavish et Graves pour le c.m-PVC asymétrique, composé des équations (4.22) jusqu'à (4.26), (4.32), (4.33) et (4.28-29), sera appelé à partir de maintenant D0. En appliquant la technique de décomposition de Benders au modèle D0, nous avons prouvé que les rayons extrémaux du polytope du dual du sous-problème linéaire associé à D0 conduisent aux contraintes (4.31) que nous avons mises au point pour le c.m-PVC asymétrique.

Supposons que les variables x_{ij} $(1 \leq i \neq j \leq n)$ et m sont fixées de façon à satisfaire aux contraintes (4.23) jusqu'à (4.29). Dénotons par $\bar{x}_{ij}$ et $\bar{m}$ ces valeurs et par D1 le problème ainsi obtenu. Les variables de flot y_{ij} $(1 \leq i \neq j \leq n)$ constituent les variables de D1 qui s'écrit:

$$\text{Min } Z = \sum_{\substack{i=1 \\ i \neq j}}^{n} \sum_{j=1}^{n} 0 \cdot y_{ij} \tag{4.37}$$

assujettie à:

(D1)

$$\sum_{\substack{j=1 \\ j \neq i}}^{n} y_{ij} - \sum_{\substack{j=1 \\ j \neq i}}^{n} y_{ji} = d_i \qquad (i = 2,\ldots,n) \tag{4.38}$$

$$y_{ij} \leq Dx_{ij} \qquad (i,j = 1,\ldots,n;\ i \neq j) \tag{4.39}$$

$$y_{ij} \geq 0 \qquad (i,j = 1,\ldots,n;\ i \neq j) \tag{4.40}$$

Soit D2 le problème dual de D1. Appelons u_i $(i = 2,\ldots,n)$ les variables duales associées aux contraintes (4.38) et z_{ij} $(i,j = 1,\ldots,n;\ i \neq j)$ les variables duales associées à (4.39). D2 s'écrit:

$$\text{Max } Z' = \sum_{\substack{i=1 \\ i \neq j}}^{n} \sum_{j=1}^{n} - Dx_{ij} z_{ij} + \sum_{i=2}^{n} d_i u_i \tag{4.41}$$

assujettie à:

(D2)

$$-z_{ij} + u_i - u_j \leq 0 \qquad (i,j = 2,\ldots,n;\ i \neq j) \tag{4.42}$$

$$-z_{1j} - u_j \leq 0 \qquad (j = 2,\ldots,n) \tag{4.43}$$

$$-z_{i1} + u_i \leq 0 \qquad (i = 2,\ldots,n) \tag{4.44}$$

$$z_{ij} \geq 0 \qquad (i,j = 1,\ldots,n;\ i \neq j) \tag{4.45}$$

$$u_i \text{ libre} \qquad (i = 2,\ldots,n) \tag{4.46}$$

Lemme 4.1

Toute solution réalisable de D1 est optimale.

preuve

Evident, car la fonction objective de D1 est la constante 0. ∎

Lemme 4.2

Si, en posant $x_{ij} = \bar{x}_{ij}$ $(i,j = 1,\ldots,n;\ i \neq j)$ et $m = \bar{m}$, la solution obtenue n'est pas réalisable pour D0, alors D1 est irréalisable[1].

preuve

La preuve est contenue dans les remarques que nous avons faites au sujet de (4.32) et (4.33). ∎

Lemme 4.3

En posant $z_{ij} = 0$ pour $i,j = 1,\ldots,n;\ i \neq j$ ainsi que $u_i = 0$ pour $i = 2,\ldots,n$, on obtient une solution réalisable pour D2. Cette solution sera appelée la solution nulle pour D2.

preuve

Evident car le membre droit des contraintes de D2 est 0. ∎

Lemme 4.4

Si D1 possède une solution réalisable alors la solution nulle de D2 est optimale pour D2.

preuve

Si D1 possède une solution réalisable alors la valeur optimale de D1 est 0, ce qui correspond à la valeur que prend (4.41) lorsque $z_{ij} = 0$ pour tout i et j et lorsque $u_i = 0$ pour tout i. Donc par la théorie de la dualité, la solution nulle pour D2 est optimale. ∎

(1) C'est-à-dire son domaine réalisable est vide.

Lemme 4.5

Soit $N_2 \subset \{2,\ldots,n\}$ et $N_1 = N-N_2$.

Définissons $\bar{z}_{ij}$ $(i,j = 1,\ldots,n;\ i\neq j)$ et $\bar{u}_i$ $(i = 2,\ldots,n)$ de la façon suivante où δ est un réel positif quelconque:

$$\bar{u}_i = \begin{cases} \delta & \text{si } i \in N_2 \\ 0 & \text{sinon} \end{cases}$$

$$\bar{z}_{1j} = 0 \quad \text{pour } j = 2,\ldots,n$$

$$\bar{z}_{i1} = \bar{u}_i \quad \text{pour } i = 2,\ldots,n$$

$$\bar{z}_{ij} = \begin{cases} \delta & \text{si } i \in N_2 \text{ et } j \in N_1 \text{ pour } i,j = 2,\ldots,n;\ i\neq j \\ 0 & \text{sinon} \end{cases}$$

En posant $z_{ij} = \bar{z}_{ij}$ pour $i,j = 1,\ldots,n$; $i \neq j$ et $u_i = \bar{u}_i$ pour $i = 2,\ldots,n$, on obtient une solution réalisable de D2 pour chaque valeur positive de δ.

Preuve

- Puisque $\delta > 0$, (4.45) et (4.43) sont satisfaites.
- Puisque $\bar{z}_{i1} = \bar{u}_i$ pour tout i, (4.44) est satisfaite.
- Soit $i,j = 2,\ldots,n$ et $i \neq j$:
  - Si $i \in N_1$ et $j \in N_1$, alors:

    $\bar{z}_{ij} = 0$, $\bar{u}_i = 0$ et $\bar{u}_j = 0$ donc (4.42) est satisfaite.
  - Si $i \in N_2$ et $j \in N_1$, alors:

    $\bar{z}_{ij} = \delta$, $\bar{u}_i = \delta$ et $\bar{u}_j = 0$ donc (4.42) est satisfaite.
  - Si $i \in N_1$ et $j \in N_2$, alors:

    $\bar{z}_{ij} = 0$, $\bar{u}_i = 0$ et $\bar{u}_j = \delta$ donc (4.42) est satisfaite.
  - Si $i \in N_2$ et $j \in N_2$, alors:

    $\bar{z}_{ij} = 0$, $\bar{u}_i = \delta$ et $\bar{u}_j = \delta$ donc (4.42) est satisfaite. ∎

Les valeurs $\bar{z}_{ij}$ et $\bar{u}_i$ dépendent de la partition particulière de N choisie (N_1 et N_2) et aussi de la valeur de δ.

Théorème 4.2

Si, en posant $x_{ij} = \bar{x}_{ij}$ $(i,j = 1,\ldots,n;\ i \neq j)$ et $m = \bar{m}$, on obtient une solution réalisable pour D0 alors la solution nulle pour D2 est optimale pour D2.

Preuve

Ce théorème découle directement du lemme 4.4. ■

Théorème 4.3

Si, en posant $x_{ij} = \bar{x}_{ij}$ $(i,j = 1,\ldots,n;\ i \neq j)$ et $m = \bar{m}$, on obtient une solution irréalisable pour D0, alors D2 est non-borné.

Preuve

Par le lemme 4.2, D1 est alors irréalisable, ce qui implique par la théorie de la dualité que D2 est non-borné. ■

Théorème 4.4

Si, en posant $x_{ij} = \bar{x}_{ij}$ $(i,j = 1,\ldots,n;\ i \neq j)$ et $m = \bar{m}$, on obtient une solution irréalisable pour D0 contenant au moins un sous-tour illégal de type 1 alors la solution réalisable pour D2 $z_{ij} = \bar{z}_{ij}$ $(i,j = 1,\ldots, n;\ i \neq j)$ et $u_i = \bar{u}_i$ $(i = 2,\ldots,n)$, obtenue en prenant N_2 comme l'ensemble des villes d'un sous-tour illégal de type 1 et $N_1 = N\text{-}N_2$, donne à l'objectif de D2 une valeur infinie lorsque $\delta = \infty$.

Preuve

L'objectif de D2 évalué avec la solution $\bar{z}_{ij}$ $(i,j = 1,\ldots,n;\ i \neq j)$ et $\bar{u}_i$ $(i = 2,\ldots,n)$ s'écrit:

$$Z' = \sum_{\substack{i=1 \\ i \neq j}}^{n} \sum_{j=1}^{n} - D\bar{x}_{ij}\, \bar{z}_{ij} + \sum_{i=2}^{n} d_i\, \bar{u}_i$$

Décomposons Z' en les 4 termes suivants:

$$Z' = \underbrace{\sum_{j=2}^{n} - D\bar{x}_{1j}\,\bar{z}_{1j}}_{A} + \underbrace{\sum_{i=2}^{n} - D\bar{x}_{i1}\,\bar{z}_{i1}}_{B} + \underbrace{\sum_{i=2}^{n}\sum_{\substack{j=2 \\ i\neq j}}^{n} - D\bar{x}_{ij}\,\bar{z}_{ij}}_{C} + \underbrace{\sum_{i=2}^{n} d_i\,\bar{u}_i}_{E}$$

On constate que:

$$A = 0 \text{ car } \bar{z}_{1j} = 0 \text{ pour } j = 2,\ldots,n$$

Puisque N_2 constitue l'ensemble des villes d'un sous-tour illégal de type 1, le graphe de la solution $\bar{m}$ et $\bar{x}_{ij}$ pour tout i,j ne contient aucun arc allant de N_2 à N_1 de même qu'aucun arc de N_1 vers N_2. On peut donc conclure que $\bar{x}_{ij} = 0$ si $(i \in N_2,\ j \in N_1)$ ou bien si $(i \in N_1,\ j \in N_2)$. On peut donc déduire que:

$$B = 0 \qquad \text{car} \begin{cases} \bar{x}_{i1} = 0 \text{ si } i \in N_2 \text{ (1)} \\ \bar{z}_{i1} = 0 \text{ si } i \in N_1 \end{cases}$$

$$C = 0 \qquad \text{car} \begin{cases} \bar{z}_{ij} = 0 \text{ si } i \in N_1 \text{ ou } j \in N_2 \\ \bar{x}_{ij} = 0 \text{ si } i \in N_2 \text{ et } j \in N_1 \end{cases}$$

$$E = \delta \cdot \sum_{i \in N_2} d_i \quad \text{car} \quad \bar{u}_i = \begin{cases} 0 \text{ si } i \notin N_2 \\ \delta \text{ si } i \in N_2 \end{cases}$$

donc: $$Z' = \delta \cdot \sum_{i \in N_2} d_i$$

et $$\lim_{\delta \to \infty} Z' = \infty$$ ■

Théorème 4.5

Si, en posant $x_{ij} = \bar{x}_{ij}$ $(i,j = 1,\ldots,n;\ i \neq j)$ et $m = \bar{m}$, on obtient

(1) On sait que $1 \in N_1$.

une solution irréalisable pour D0 contenant au moins un sous-tour illégal de type 2 alors la solution réalisable pour D2 $z_{ij} = \bar{z}_{ij}$ $(i,j = 1,\ldots,n;\ i \neq j)$ et $u_i = \bar{u}_i$ $(i = 2,\ldots,n)$, obtenue en prenant N_2 comme l'ensemble des villes d'un sous-tour illégal de type 2 et $N_1 = N - N_2$, donne à l'objectif de D2 une valeur infinie lorsque $\delta = \infty$.

Preuve

Evaluons la valeur, notée Z', que donne à l'objectif de D2 la solution réalisable $z_{ij} = \bar{z}_{ij}$ pour tout i et j et $u_i = \bar{u}_i$ pour tout i:

$$Z' = \underbrace{\sum_{j=2}^{n} - D\bar{x}_{1j}\,\bar{z}_{1j}}_{A} + \underbrace{\sum_{i=2}^{n} - D\bar{x}_{i1}\,\bar{z}_{i1}}_{B} + \underbrace{\sum_{i=2}^{n}\sum_{\substack{j=2 \\ i\neq j}}^{n} - D\bar{x}_{ij}\,\bar{z}_{ij}}_{C} + \underbrace{\sum_{i=2}^{n} d_i\,\bar{u}_i}_{E}$$

On constate que:

$A = 0$ car $\bar{z}_{1j} = 0$ pour $j = 2,\ldots,n$

$B = -D\cdot\delta$ car $\begin{cases} \bar{z}_{i1} = 0 \text{ si } i \in N_1 \\ \bar{x}_{i1} = 0 \text{ si } i \in N_2 \text{ et } i \neq \bar{i} \\ \bar{x}_{\bar{i}1} = 1 \text{ et } \bar{z}_{\bar{i}1} = \bar{u}_{\bar{i}} = \delta \end{cases}$

Note: $\bar{i} \in N_2$ est la dernière ville visitée par le véhicule avant de revenir au dépôt.

$C = 0$ car $\begin{cases} \bar{z}_{ij} = 0 \text{ si } i \in N_1 \text{ ou } j \in N_2 \\ \bar{x}_{ij} = 0 \text{ si } i \in N_2,\ j \in N_1 \text{ et } i,j \geq 2 \end{cases}$

$E = \delta \cdot \sum_{i \in N_2} d_i$ car $\bar{u}_i = \begin{cases} 0 \text{ si } i \in N_2 \\ \delta \text{ si } i \in N_2 \end{cases}$

on peut donc écrire:

$$Z' = -D\cdot\delta + \delta\cdot\sum_{i\in N_2} d_i = \delta\cdot\left(\left[\sum_{i\in N_2} d_i\right] - D\right)$$

Puisque par hypothèse N_2 est l'ensemble des villes d'un sous-tour illégal de type 2, on a:

$$\left[\sum_{i \in N_2} d_i\right] - D > 0$$

donc

$$\lim_{\delta \to \infty} Z' = \infty.$$

■

Dans les théorèmes 4.4 et 4.5 on considère des solutions $\bar{x}_{ij}$ et $\bar{m}$ de D_0 (modèle de Gavish et Graves) qui possèdent les 3 propriétés suivantes:

1. elles sont entières;
2. elles satisfont aux contraintes de degré sur les sommets;
3. elles sont associées à un graphe contenant au moins un sous-tour illégal de type 1 (théorème 4.4) ou de type 2 (théorème 4.5).

Dans ces deux théorèmes, on démontre qu'à partir de telles solutions, on peut trouver des rayons extrémaux de D2, le dual du problème linéaire D1 associé à D0. Plus précisément, chaque sous-tour illégal de type 1 ou de type 2 conduit à un rayon extrémal qui à son tour permet d'établir une contrainte valide sur les variables de D0. Le théorème 4.6 prouve que la forme de cette dernière contrainte correspond à une des contraintes de (4.31) que nous avons mise au point pour le c.m-PVC asymétrique.

Théorème 4.6

Soit $x_{ij} = \bar{x}_{ij}$ $(i,j = 1,\ldots,n;\ i \neq j)$ et $m = \bar{m}$ une solution irréalisable de D0 telle que les 3 propriétés précédentes sont satisfaites. Soit $z_{ij} = \bar{z}_{ij}$ $(i,j = 1,\ldots,n;\ i \neq j)$, $u_i = \bar{u}_i$ $(i = 2,\ldots,n)$ avec N_2 l'ensemble des villes d'un sous-tour illégal de type 1 ou de type 2 et N_1 $N-N_2$. En posant $\delta = 1$, l'inégalité suivante est valide pour D0[1]:

(1) Et donc pour le c.m-PVC asymétrique.

$$Z'(\bar{z}_{ij}, \bar{u}_i)^{(1)} \leq 0$$

De plus cette inégalité est équivalente à la contrainte (4.31) que nous avons établie pour le c.m-PVC asymétrique.

Preuve

Par les théorèmes 4.4 et 4.5, on sait que le vecteur des valeurs $\bar{z}_{ij}$ et $\bar{u}_i$ constitue un rayon extrémal pour le polytope de D2. Pour que D2 soit borné et D1 réalisable, les variables x_{ij} doivent donc satisfaire à la contrainte[(2)]:

$$Z'(\bar{z}_{ij}, \bar{u}_i) \leq 0 \;\Leftrightarrow$$

$$\sum_{i=1}^{n} \sum_{\substack{j=1 \\ i \neq j}}^{n} -D x_{ij} \bar{z}_{ij} + \sum_{i=2}^{n} d_i \bar{u}_i \leq 0 \;\Leftrightarrow$$

$$\sum_{i=1}^{n} \sum_{\substack{j=1 \\ i \neq j}}^{n} x_{ij} \bar{z}_{ij} \geq \frac{\sum_{i=2}^{n} d_i \bar{u}_i}{D} \;\Leftrightarrow$$

$$\underbrace{\sum_{j=2}^{n} x_{1j} \bar{z}_{1j}}_{A} + \underbrace{\sum_{i=2}^{n} x_{i1} \bar{z}_{i1}}_{B} + \underbrace{\sum_{i=2}^{n} \sum_{\substack{j=2 \\ i \neq j}}^{n} x_{ij} \bar{z}_{ij}}_{C} \geq \underbrace{\frac{\sum_{i=2}^{n} d_i \bar{u}_i}{D}}_{E}$$

On constate que:

$A = 0$ car $z_{1j} = 0$ pour $j = 2, \ldots, n$

$B = \sum_{i \in N_2} x_{i1}$ car $\begin{cases} \bar{z}_{i1} = \bar{u}_i = 0 \text{ si } i \in N_1 \\ \bar{z}_{i1} = \bar{u}_i = \delta = 1 \text{ si } i \in N_2 \end{cases}$

---

(1) Z' symbolise la fonction objective de D2 et $Z'(\bar{z}_{ij}, \bar{u}_i)$ signifie que Z' est évaluée en $z_{ij} = \bar{z}_{ij}$ et $u_i = \bar{u}_i$. Notons que les x_{ij} sont variables dans ce théorème.

(2) Le lecteur est ici référé à [9] concernant la théorie de la décomposition de Benders pour plus de détails.

$$C = \sum_{i \in N_2} \sum_{j \in N_1} x_{ij} \quad \text{car} \quad \begin{cases} \bar{z}_{ij} = 0 \text{ si } i \in N_1 \text{ ou } j \in N_2 \\ \bar{z}_{ij} = \delta = 1 \text{ si } i \in N_2 \text{ et } j \in N_1 \end{cases}$$

$$i,j \geq 2$$

$$E = \frac{\sum_{i \in N_2} d_i}{D} \quad \text{car} \quad \begin{cases} \bar{u}_i = 0 \text{ si } i \in N_1 \\ \bar{u}_i = \delta = 1 \text{ si } i \in N_2 \end{cases}$$

Donc les variables x_{ij} pour le c.m-PVC asymétrique doivent satisfaire à:

$$\sum_{i \in N_2} x_{i1} + \sum_{i \in N_2} \sum_{j \in N_1} x_{ij} \geq \frac{\sum_{i \in N_2} d_i}{D}$$

$$i,j \geq 2$$

ce qui s'écrit:

$$\sum_{i \in N_2} \sum_{j \in N_1} x_{ij} \geq \frac{\sum_{i \in N_2} d_i}{D}$$

Puisque le terme de gauche de l'inégalité précédente est entier, on peut écrire:

$$\sum_{i \in N_2} \sum_{j \in N_1} x_{ij} \geq \left\lceil \frac{\sum_{i \in N_2} d_i}{D} \right\rceil$$

ce qui est la contrainte (4.31) avec $S = N_2$. ∎

4.5 Résumé de la littérature concernant les méthodes de résolution du c.m-PVC

4.5.1 Méthodes approximatives pour le c.m-PVC[1]

Le c.m-PVC est un problème extrêmement difficile à résoudre opti-

(1) Plusieurs des algorithmes dont il sera mention ici peuvent être utilisés pour résoudre des modèles plus complexes que le c.m-PVC où des contraintes de temps maximum, des contraintes de fenêtre, etc. sont aussi imposées sur les tours. Nous en ferons la description dans le contexte du c.m-PVC.

malement. Pour préciser ces termes, mentionnons que présentement le PVC peut être résolu de façon exacte en un temps raisonnable sur ordinateur pour des problèmes contenant environ 300 villes dans le cas asymétrique[1] et de l'ordre de 120 villes dans le cas symétrique[2]. Par contre, à notre connaissance, les c.m-PVC de plus grande taille résolus optimalement contiennent 30 villes[3].

La majorité des méthodes construites pour le c.m-PVC fournissent donc une solution approximative. On compte dans la littérature un très grand nombre d'algorithmes heuristiques adaptés au c.m-PVC. Il n'est pas dans notre intention ici d'en effectuer un résumé exhaustif puisque nos travaux portent sur la construction de méthodes exactes pour ce problème. Nos remarques les concernant seront donc brèves et un seul d'entre eux sera décrit ici.

Le plus célèbre heuristique mis au point pour le c.m-PVC est indubitablement celui de Clarke et Wright [25]. Il est implanté dans plusieurs codes commerciaux et a reçu une grande attention de la part des chercheurs. Le fameux "savings" de Clarke et Wright ont été généralisés à des modèles plus complexes que le c.m-PVC, adaptés à des cas particuliers du c.m-PVC ou même supposément améliorés par l'ajout de certains paramètres. La littérature à leur sujet est abondante et Harry O'Brien [97] dans une lettre à l'éditeur de la revue "Operational Research Quarterly", se montre à tout le moins critique vis-à-vis l'utilité pratique de tels articles. Nous éviterons de nous immiscer dans une telle polémique. On peut assurément affirmer que l'heuristique de Clarke et Wright est rapide et facile d'implantation. C'est pourquoi nous avons choisi cette méthode pour calculer une borne supérieure initiale dans notre méthode de "Branch-and-Bound" que nous avons construite pour le c.m-PVC. Nous la décrirons ici.

Dans sa version originale l'heuristique de Clarke et Wright tient

---

(1) Voir 2.3.2, algorithme de Balas et Christofides [2].

(2) Voir 2.6.3.

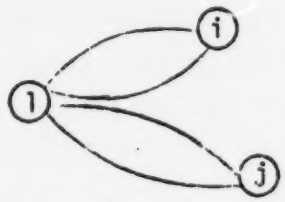
(3) Il s'agit de c.m-PVC symétriques résolus par notre algorithme.

compte de véhicules de capacités différentes, C est asymétrique euclidienne et les $m^{(1)}$ tours sont construits simultanément[2]. Nous supposerons que les véhicules ont une capacité de livraison identique, l'idée sur laquelle reposent les "savings" demeurant inchangée. La méthode Clarke et Wright débute avec la solution où chacune des villes est desservie par un véhicule différent. Cette solution est graduellement améliorée en insérant sur un même tour des villes distinctes à l'aide de la notion de savings" (s_{ij}) dont le calcul est illustré dans la figure 4.2.

FIGURE 4.2

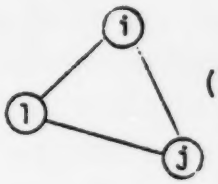
Calcul des "savings" de Clarke et Wright

A: i et j desservies séparément



coût: $2c_{1i} + 2c_{1j}$

B: i et j sur le même tour



coût : $c_{1i} + c_{ij} + c_{1j}$

Profit d'utiliser B au lieu de A = $s_{ij} = (2c_{1i} + 2c_{1j}) - (c_{1i} + c_{ij} + c_{1j})$

$$\boxed{s_{ij} = c_{1i} + c_{1j} - c_{ij}} \quad (3)$$

Un tableau des "savings" triés par ordre décroissant est calculé initialement. Chaque "saving" est par la suite analysé exactement une

(1) Au début de la méthode, la valeur que prendra m est inconnue.

(2) La méthode peut être modifiée pour construire un tour à la fois.

(3) $s_{ij} \geq 0$ car C est euclidienne. Si tel n'est pas le cas, certains "savings" pourraient être négatifs. Le calcul de s_{ij} dans le cas asymétrique est similaire et conduit à (4.36).

fois en partant du plus élevé et l'arête associée au "saving" sous considération, disons s_{ij}, est insérée dans la solution présente si un seul véhicule peut desservir la route dont faisait partie i ainsi que la route sur laquelle se trouvait j. Notons que s_{ij} n'est pas considéré si i ou j se trouve sur le "milieu"(1) d'une route; de plus, si l'arête (i,j) est insérée, les arêtes (1,i) et (1,j) sont éliminées de la solution en main.

La méthode de Clarke et Wright ainsi que plusieurs autres reposent sur le calcul de "savings" associés à l'insertion d'arêtes dans une solution. D'autres heuristiques, notamment celui de Hays [57], évaluent des "savings" concernant l'insertion d'une ville dans un tour. En général, selon Christofides [16], de telles méthodes sont rapides mais fournissent des solutions de moins bonne qualité que des méthodes plus lentes basées sur la transformation du m-PVC restreint(2) qu'est le c.m-PVC en un PVC où les chaînes (ou chemins) entre deux villes artificielles sont assujetties (s) aux contraintes de capacité. En utilisant un heuristique 2-opt. [84] ou 3-opt. sur une solution réalisable d'un tel PVC, il semble que les solutions obtenues soient excellentes (voir [18]) et les méthodes utilisant des "savings" peuvent fournir une solution initiale à de tels heuristiques.

Récemment Fisher et Jaikumar [36] ont décrit un nouvel heuristique pour le c.m-PVC. Ceux-ci mentionnent que la faiblesse de la plupart des méthodes approximatives concernant ce problème est qu'elles ne tiennent pas suffisamment compte des contraintes de capacité. En relaxant les contraintes sur les sous-tours illégaux de type 1 dans leur formulation du c.m-PVC, ils en arrivent à un problème d'affectation généralisée où l'on doit assigner les villes aux véhicules tout en respectant les capacités (possiblement différentes) de ceux-ci. Les coûts utilisés dans cette relaxation du c.m-PVC sont établis de façon à approximer le coût des PVC obtenus pour une affectation particulière des villes

---

(1) Liée à deux autres villes différentes du dépôt.

(2) Un m-PVC auquel on adjoint des contraintes de capacité.

aux véhicules. Une fois ce problème d'affectation généralisée résolu, des PVC sont résolus (de façon heuristique ou optimale) pour chacun des ensembles de villes auquel un véhicule a été affecté. Les tests de performance effectués par Fisher et Jaikumar indiquent que leur méthode est aussi rapide que celle de Clarke et Wright et fournit des solutions d'une qualité de beaucoup supérieure.

4.5.2 Résumé des principales méthodes exactes de résolution du c.m-PVC(1)

Comparativement au grand nombre d'heuristiques pour le c.m-PVC publiés dans la littérature, relativement peu de méthodes exactes ont été proposées pour ce problème. A partir de la formulation du c.m-PVC de Balinski et Quandt, décrite dans la section 4.3.1 de ce chapitre, Foster et Ryan [38] ont développé une approche proposée par Pierce [100] où l'on tient compte de contraintes supplémentaires(2) dans la génération des tours réalisables. Ce faisant, le nombre de variables x_j dans le modèle est de beaucoup diminué et ce dernier peut être résolu optimalement. Le résultat obtenu est toutefois approximatif mais de bonne qualité par rapport à [18,42,49,126] selon les tests effectués.

Dans un même ordre d'idées, Rao et Zionts [103] ont suggéré de résoudre le modèle de Balinski et Quandt avec une technique de génération de colonnes [77]. Leur article ne contient cependant aucun résultat de tests sur ordinateur.

Une méthode basée sur la génération des tours réalisables du c.m-PVC auquel est adjoint plusieurs autres contraintes a été aussi construite par Christofides [16]. La solution optimale d'un problème contenant 31 villes a été obtenue avec cette méthode. Soulignons que

---

(1) Encore ici la plupart d'entre elles ont été construites pour résoudre des modèles plus complexes que le c.m-PVC.

(2) Autres que la restriction de capacité.

dans cet article Christofides mentionne à la décharge des méthodes optimales pour le c.m-PVC que, même si elles ne peuvent traiter des problèmes de taille restreinte (environ 30 villes) par rapport aux heuristiques, plusieurs situations pratiques sont réductibles à de telles tailles[1]. En outre, plusieurs méthodes optimales, même si elles ne sont pas complétées, peuvent s'avérer utiles lorsque utilisées en conjonction avec un heuristique.

Dans la méthode de Christofides, les tours réalisables ne sont pas engendrés a priori mais plutôt au besoin dans un arbre du type LIFO. Une telle méthode peut facilement tenir compte d'un grand nombre de contraintes de différents types sur les tours et d'une certaine façon plus les tours seront assujettis à satisfaire à plusieurs restrictions moins leur nombre sera grand et plus efficace sera la méthode d'énumération.

Récemment Christofides, Mingozzi et Toth [21] ont développé trois nouveaux algorithmes optimaux pour le c.m-PVC où m est fixé à l'avance et où, en plus des contraintes de capacité, des contraintes de coût total maximum sont imposés sur les tours[2]. En utilisant les mêmes variables que dans le modèle de Golden et al. (4.3.3), les contraintes s'écrivent:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}^{k} + \sum_{i=2}^{n} u_i \left(\sum_{j=1}^{n} x_{ij}^{k} \right) \leq T \qquad (4.47)$$

$$\text{pour } k = 1,\ldots,\bar{m}$$

où u_i $(i = 2,\ldots,n)$ est le coût de déchargement des biens demandés à la ville i.

Le modèle de Golden et al. augmenté de (4.47) est évidemment trop complexe pour être résolu directement. Les trois algorithmes de Christofides, Mingozzi et Toth seront décrits ici de façon brève.

---

(1) Dans [16] on cite entre autres les problèmes concernant la livraison de matériaux pour la construction, de fertilisants chimiques et aussi de pièces d'automobile.

(2) Il ne s'agit donc plus d'un c.m-PVC selon la définition que nous avons donné à ce problème.

Ces trois algorithmes sont basés sur le calcul de deux bornes inférieures différentes pourle problème considéré. La première est obtenue en spécialisant une borne inférieure valide pour le m-PVC au cas de leur problème. Le calcul de cette borne repose sur la notion d'un m-arbre et est une extension des travaux de Held et Karp [60] pour le PVC et ceux de Gavish et Srikanth [47] pour le m-PVC.

La deuxième borne inférieure analysée par Christofides et al. repose sur la notion d'un q-chemin. Une notion similaire, celle d'un n-chemin, a été introduite par Houck et al. [65] pour le PVC. Un q-chemin(1) est un chemin de la forme $(1, i_1, \ldots, i_k)$ où $k \geq 1$, $i_\ell \neq 1$ pour $1 \leq \ell \leq k$ et où $\sum_{\ell=1}^{k} d_{i_\ell} = q$. On ne suppose toutefois pas que les i_ℓ sont tous différents. Par extension, on peut définir un q-tour comme un q-chemin auquel on ajoute l'arc $(i_k,1)$. Un q-tour est donc un tour dont les villes ont une demande totale de q et où on permet qu'une ville soit visitée plus d'une fois(2).

D'après Christofides et al. [21], la borne inférieure obtenue à partir des q-tours est supérieure à la première borne qu'ils ont testée pour leur problème. A l'aide de leurs algorithmes, ils ont résolu optimalement 10 problèmes dont la taille variait entre 10 et 25 villes.

4.6 Une nouvelle méthode optimale pour le c.m-PVC

4.6.1 Description générale

Notre algorithme pour le c.m-PVC est basé en général sur un processus de relaxation-réintroduction de contraintes similaire à celui que nous avons utilisé dans notre algorithme pour le m-PVC. Nos modèles pour ces deux problèmes sont presque identiques et la seule différence au niveau de leur formulation réside dans le fait que $\min_S = 1$ pour tout

(1) Dans le cas symétrique, on pourrait parler d'une q-chaîne.

(2) Il ne s'agit donc pas d'un chemin simple.

$S \subset N-\{1\}$ dans le cas du m-PVC tandis que $\min_S \geq 1$ s'il s'agit du c.m-PVC. Il est donc tout à fait naturel, étant donné le succès que nous avons obtenu sur le m-PVC, de tenter de résoudre le c.m-PVC de façon similaire.

Notre méthode de résolution du c.m-PVC s'applique de façon identique au modèle P2 vu en 4.3.2 et au modèle P4 donné en 4.4.2 que nous avons mis au point pour, respectivement, le c.m-PVC symétrique et asymétrique. Dans le but de faciliter la description de notre algorithme, nous supposerons toutefois que C est symétrique tout au long de cette section. Notre modèle P2 peut s'écrire:

$$\text{Min } Z = \sum_{i<j} c_{ij} x_{ij} \tag{4.48}$$

assujettie à:

Contraintes de degré

$$\sum_{j=2}^{n} x_{1j} = 2m \tag{4.49}$$

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2 \qquad (k = 2,\ldots,n) \tag{4.50}$$

Contraintes d'élimination de sous-tours illégaux

$$\sum_{\substack{i\in S, j\in S \\ i<j}} x_{ij} \leq |S| - \min_S \qquad (S \subset N-\{1\}) \tag{4.51}$$

Contraintes de définition sur les variables

$$0 \leq x_{ij} \leq 1 \qquad (2 \leq i<j \leq n) \tag{4.52}$$

$$0 \leq x_{1j} \leq 2 \qquad (j = 2,\ldots,n) \tag{4.53}$$

$$x_{ij} \text{ entier} \qquad (1 \leq i<j \leq n) \tag{4.54}$$

$$m \geq 0 \text{ et entier}^{(1)}$$

Le très grand nombre de contraintes dans (4.51)[2] rend P2 extrêmement difficile à résoudre directement même pour n petit. Pour y arriver, on procède plutôt de la façon générale suivante: les contraintes d'éli-

(1) m sera entière si les autres variables sont entières.

(2) Dans (4.51) la cardinalité de s varie de 3 à n-2.

mination de sous-tours illégaux (4.51) ainsi que les contraintes d'intégrité (4.54) sont initialement relaxées et le modèle linéaire obtenu est résolu par la méthode du simplex. Si la solution trouvée n'est pas entière, l'intégrité est restaurée en utilisant une méthode de "Branch-and-Bound"; si par contre la solution continue en main est entière, on vérifie si les contraintes de (4.51) sont satisfaites. Dans la négative, certaines contraintes de (4.51) sont adjointes explicitement au modèle[1] et dans l'affirmative une nouvelle solution réalisable pour le c.m-PVC est détectée. Il faut alors choisir un nouveau sous-problème à traiter puisqu'il s'agit d'une méthode de "Branch-and-Bound". Notons que la première solution réalisable trouvée n'est habituellement pas optimale puisque certaines variables du problème ont été fixées à des valeurs entières si on se trouve à un niveau autre que le niveau 0 dans l'arbre de l'énumération. Notre algorithme pour le m-PVC, utilisant des coupes de Gomory pour restaurer l'intégrité, n'a pas ce désavantage mais peut cependant, s'il n'est pas complété[2], se terminer sans que ni la solution optimale ni aucune solution réalisable n'ait été découverte. Une méthode de coupes a été testée pour maintenir l'intégrité dans le cas du c.m-PVC. Toutefois la convergence n'a été obtenue que pour un petit nombre de problèmes de 10 et 15 villes. Dans la plupart des tests sur ordinateur, le déterminant de la base devenait trop grand pour que l'ajout de nouvelles coupes soit efficace.

Dans le processus précédemment décrit, il a été décidé que les contraintes de degré seront toutes présentes initialement dans le problème relaxé même si à la rigueur quelques-unes d'entre elles peuvent être aussi relaxées. Dans la solution optimale de la plupart des m-PVC que nous avons résolus avec une telle approche, ces contraintes étaient presque toutes effectives et il nous semblait peu

---

(1) Evidemment (4.11), forme équivalente de (4.51), pourrait aussi être utilisée. Nous avons vu au chapitre 3 de ce texte pourquoi nous préférons la forme des contraintes d'élimination de sous-tours illégaux qui porte sur des variables x_{ij} avec i et j dans S plutôt que celle associée aux variables reliant S et $\bar{S}$.

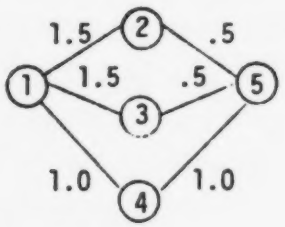
(2) La cause la plus fréquente est le déterminant de la base qui devient trop grand (voir chapitre 3).

probable qu'un tel phénomène ne se reproduise pas dans le cas du c.m-PVC. En fait, après l'exécution de tests sur ordinateur, nous avons constaté que notre intuition s'est avérée justifiée.

Il est possible en théorie, même avec une méthode de "Branch-and-Bound", de procéder de façon analogue à l'algorithme RENVERSE que nous avons construit pour le m-PVC, c'est-à-dire de satisfaire aux contraintes d'élimination de sous-tours illégaux avant celles sur l'intégrité des variables. Toutefois, pour le c.m-PVC, ceci n'est pas valide comme le démontre le contre-exemple suivant:

FIGURE 4.3

Une méthode du type RENVERSE n'est pas valide pour le c.m-PVC



$S = \{2,3,4,5\}$

$D = 10$

$d_2 = d_3 = d_4 = d_5 = 3$

Supposons que par un certain processus d'étiquetage, la composante connexe illustrée dans la figure 4.3 a été identifiée. Les contraintes de degré sont satisfaites par les sommets de celle-ci et le dépôt fait partie de cette composante; il ne s'agit donc pas d'un sous-tour illégal de type 1, le terme sous-tour étant étendu ici au cas où les variables associées ne sont pas nécessairement entières. Toutefois la demande totale des villes 2,3,4 et 5 (12) dépasse la capacité d'un véhicule (10) et la figure 4.3 représente un sous-tour illégal de type 2. Supposons qu'on utilise la contrainte de (4.51) basée sur $S = \{2,3,4,5\}$ pour éliminer ce sous-tour: on obtient $\sum_{i\in S, j\in S, i<j} x_{ij} = 2$ et $|S| - \min_S = 2$; un tel sous-

tour avec variables fractionnaires ne sera donc pas éliminé par (4.51). La raison de ce phénomène repose sur le fait que $\sum_{j \in S} x_{1j} > 2$ et ainsi il y a plus d'un véhicule qui parcourt S. Avec $\sum_{j \in S} x_{1j}$ une solution entière, toute composante connexe contenant le dépôt et identifiée par le processus d'étiquetage sera telle que $\sum_{j \in S} x_{1j} = 2$ si $S \subset N-\{1\}$ est l'ensemble de ses sommets. Ainsi celle-ci pourra être éliminée par (4.51) dans le cas où la demande de ses villes dépasse la capacité d'un véhicule. Remarquons que pour une composante connexe sur $S \subset N-\{1\}$ ne contenant pas le dépôt, $\sum_{\substack{i \in S, j \in S \\ i<j}} x_{ij} = |S|$[1] et (4.51) pourrait servir à l'éliminer si nécessaire, même avec une solution fractionnaire. Il nous semble toutefois inutile d'éliminer de telles composantes connexes avant que l'intégrité soit atteinte car de toute façon le processus d'étiquetage devra être exécuté aussi avec une solution entière. Ce processus est similaire à celui développé dans notre algorithme pour le m-PVC à la nuance près que maintenant les différentes composantes connexes contenant le dépôt doivent aussi être identifiées[2] et analysées.

4.6.2 Implantation sur ordinateur

Nous avons modifié le programme BB[3] de Land et Powell [71] de façon à l'adapter à la méthode de résolution du c.m-PVC que nous venons de décrire. Après quelques essais sur ordinateur, il est apparu évident que nous ne pourrions atteindre à l'aide de notre processus de relaxation des tailles aussi élevées sur le c.m-PVC que sur le m-PVC. Dans le but d'accroître l'efficacité de notre méthode de "Branch-and-Bound" nous y avons donc ajouté plusieurs raffinements techniques. L'organigramme de la figure 4.4 les contient et résume les commentaires précédents. Nous

---

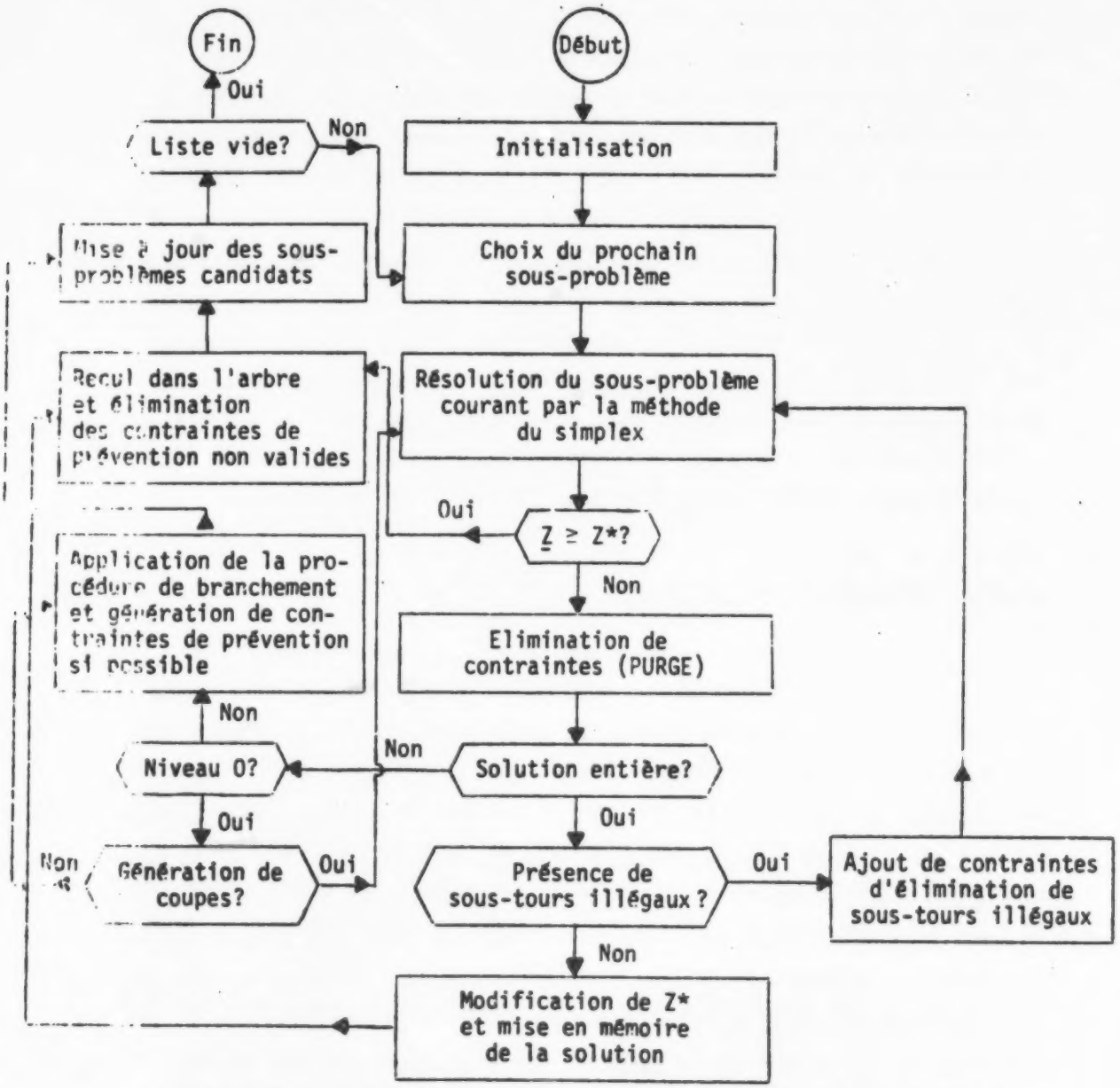
(1) Ce qui n'est pas nécessairement le cas lorsque la composante connexe contient de dépôt (voir figure 4.3).

(2) Dans le cas du m-PVC, le dépôt et les sous-tours qui lui sont reliés constituent après l'exécution du processus d'étiquetage une seule composante connexe; ce qui n'est pas le cas ici. Des modifications mineures doivent être apportées à la sous-routine d'étiquetage pour traiter de ce phénomène.

(3) Voir 1.6.1 pour une description de ce programme.

FIGURE 4.4

Organigramme de notre algorithme pour le c.m-PVC



Note: Z* = coût de la meilleure solution réalisable en main. Initialement la valeur de Z* est calculée à l'aide de l'heuristique de Clarke et Wright.

$\underline{Z}$ = coût de la solution continue du sous-problème courant (= ∞ si celui-ci est irréalisable).

ne décrirons ici que les changements majeurs apportés au programme BB.

Analyse de la présence de sous-tours illégaux

La première modification importante apportée au programme BB survient à la branche "oui" du test sur l'intégrité de la solution en main. Dans un algorithme de "Branch-and-Bound" ordinaire, on peut passer tout de suite à la modification de Z*. Toutefois ici il faut vérifier tout d'abord si la solution considérée satisfait aux contraintes qui ont été relaxées. Ce test ainsi que l'ajout de contraintes d'élimination de sous-tours illégaux si nécessaire sont effectués dans une sous-routine unique, appelée encore ici VERTEX, que nous avons adjointe au programme BB. VERTEX constitue en fait la codification en langage FORTRAN d'un processus d'étiquetage similaire à celui que nous avons décrit pour le m-PVC mais modifié de façon à différencier les sous-tours qui contiennent le dépôt. VERTEX effectue aussi l'ajout de contraintes de (4.51)[1] selon la politique suivante: une contrainte de (4.51) est insérée dans le modèle pour chaque sous-tour illégal de type 1 ou de type 2 détecté. Cette politique s'est avérée efficace dans le cas du m-PVC et n'a pas été modifiée ici.

Ajout de coupes de Gomory

Si la solution en main n'est pas entière, nous avons modifié BB de façon à ce que des coupes de Gomory puissent être ajoutées dans le modèle au niveau "0" de l'arbre et ceci dans le but principal d'augmenter la valeur de $\underline{Z}$ tout en évitant possiblement un ou plusieurs branchements. La sous-routine INTCON calcule pour une solution fractionnaire une seule coupe de Gomory et l'ajoute au modèle si elle est jugée suffisamment forte selon un certain critère de profondeur[2]. Dans le cas contraire, on passe aussitôt à la procédure de branchement. No-

(1) (4.30) dans le cas asymétrique.

(2) La sous-routine INTCON a été décrite en 2.6.2.

tons que nous permettons la présence d'un maximum de 5 coupes de Gomory dans le problème à des fins d'économie d'espace-mémoire. Le chiffre 5 a été choisi arbitrairement mais en pratique dans tous nos tests, il n'a jamais été atteint car pour le c.m-PVC les coupes de Gomory deviennent inefficaces rapidement(1) et la procédure de branchement était toujours utilisée très tôt dans le processus. Evidemment il serait possible d'adjoindre dans le modèle des coupes à un niveau quelconque de l'arbre. Leur validité serait toutefois dépendante du niveau auquel elles seraient ajoutées(2) et il faudrait dépenser du temps d'ordinateur pour les éliminer lorsqu'un niveau plus bas que celui auquel elles ont été insérées dans le problème est atteint. Il nous a semblé inutile vu leur inefficacité pour le c.m-PVC de procéder ainsi.

Contraintes de prévention

On note dans le rectangle de l'organigramme concernant la procédure de branchement qu'on y fait mention de contraintes dites de "prévention". Nous avons mis au point de telles contraintes dans le but principal de prévenir le plus possible la formation de sous-tours illégaux et minimiser ainsi le nombre de contraintes d'élimination de tels sous-tours qui devront être adjointes au problème. De plus leur ajout explicite dans le modèle peut augmenter $\underline{Z}$ et conduire à l'élimination rapide d'un sous-problème donné. Pour construire les contraintes de prévention, une structure de données décrivant l'état des différentes chaînes(3) formées par la fixation de variables à la valeur 1 est mise à jour tout au long de l'énumération. Pour décrire ces chaînes, on a essentiellement besoin de 3 informations: leur sommet initial, leur sommet terminal et leur poids défini comme la demande totale des villes qui en font partie. Dans le cas symétrique, on suppose que l'in-

(1) Ceci est directement relié au fait qu'avec notre algorithme où des coupes sont utilisées, le déterminant de la base augmentait très rapidement dans tous nos tests sur le c.m-PVC.

(2) Des coupes ajoutées au niveau 0 sont toujours valides au cours de l'énumération.

(3) Chemins dans le cas asymétrique.

dice du sommet débutant une chaîne est plus petit que celui du sommet la terminant. Au niveau 0 de l'arbre on considère qu'il y a n chaînes dont les sommets initiaux et terminaux coïncident. On doit noter aussi, pour chaque ville, l'étiquette (ou numéro) de la chaîne dont elle fait partie(1). Lorsqu'une variable, disons x_{ij}, est fixée à la valeur 1 à un niveau quelconque de l'arbre, la sous-routine CHCONFIX est appelée. Celle-ci est divisée principalement en deux étapes:

1. modification de la structure de données concernant les chaînes affectées par la fixation de x_{ij} à la valeur 1;
2. analyse de la génération possible d'une contrainte de prévention.

Considérons tout d'abord la première étape et analysons le cas le plus simple où i = 1. Il faut s'assurer que les chaînes débutant au sommet 1 sont différenciées de façon à ce que chaque chaîne identifiée par la structure de données soit obligatoirement parcourue par un seul véhicule. Ainsi dans le cas où une variable x_{1j} est fixée à la valeur 1, si DEB_j(2) et FIN_j(2) sont respectivement le sommet initial et le sommet terminal de la chaîne dont fait partie la ville j, on note dans la structure de données que:

1. $DEB_j = 1$ si avant que x_{1j} soit fixée à 1 on avait $DEB_j \neq 1$;
2. $FIN_j = 1$ sinon.

Cette deuxième possibilité signifie qu'en fixant x_{1j} à 1 la chaîne résultante constitue en fait un cycle. La figure 4.5 résume les commentaires précédents.

FIGURE 4.5 / PAGE SUIVANTE

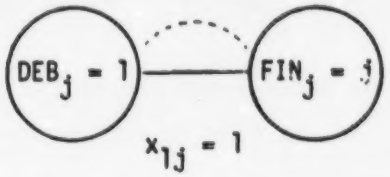
(1) La mise à jour de ces chaînes est similaire au processus d'étiquetage.

(2) A cause des contraintes de degré on a clairement que $DEB_j = j$ ou bien $FIN_j = j$ sinon le sommet j serait au milieu de la chaîne dont le début est DEB_j et dont la fin est FIN_j; en fixant x_{jj} à la valeur 1, le degré du sommet j serait 3 et le problème obtenu serait irréalisable.

FIGURE 4.5

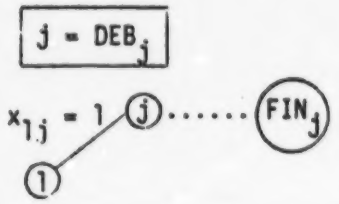
Modification de la structure de données si x_{1j} est fixée à 1

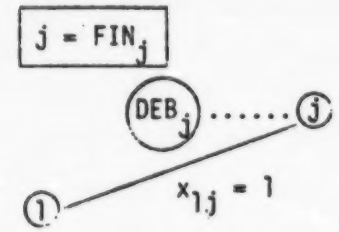
$DEB_j = 1$ avant la fixation de x_{1j}



Donc FIN_j devient 1

$DEB_j \neq 1$ avant la fixation de x_{1j}





Donc FIN_j devient DEB_j

Dans la figure 4.5, la chaîne contenant la ville j avant la fixation de x_{1j} est illustrée en pointillées. Evidemment, le cas où $DEB_j = FIN_j = j$ est possible et fait partie implicitement de la partie droite de la figure 4.5.

Pour traiter du cas où $i \neq 1$ dans la première étape de CHCONFIX, il faut tenir compte de plusieurs possibilités. La figure 4.6 illustre celles-ci avec les conventions suivantes:

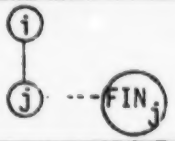
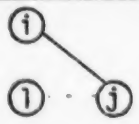
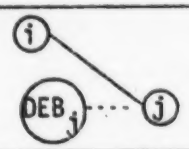
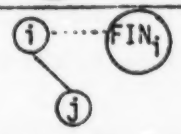
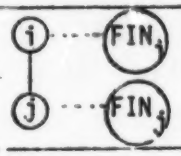
- C_k dénote la chaîne dont fait partie la ville k.
- Si $DEB_k = FIN_k$ on représente alors C_k simplement par le sommet k sinon on place DEB_k à gauche des pointillées.
- DEB et FIN dénotent respectivement le sommet initial et le

sommet terminal de la chaîne résultante à la fixation de x_{ij} à la valeur 1.

- $\min(p,\ell) = \begin{cases} p & \text{si } p < \ell \\ \ell & \text{sinon} \end{cases}$
- $\max(p,\ell) = \begin{cases} p & \text{si } p > \ell \\ \ell & \text{sinon} \end{cases}$
- Si k = 1, on indique explicitement dans la figure 4.6 la valeur 1.

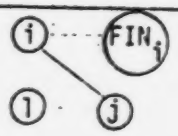
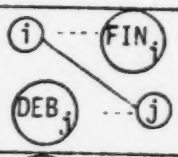
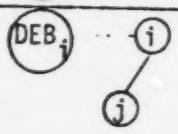
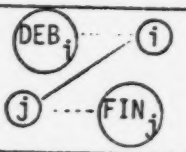
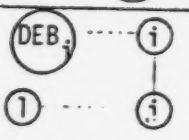
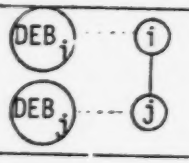
FIGURE 4.6

Modification de la structure de données si x_{ij} est fixé à 1 et si $i \neq 1$

1)	C_i : C_j :		⇒	DEB = i FIN = j
2)	C_i : C_j :		⇒	DEB = i FIN = FIN$_j$
3)	C_i : C_j :		⇒	DEB = 1 FIN = i
4)	C_i : C_j :		⇒	DEB = min (i,DEB$_j$) FIN = max (i,DEB$_j$)
5)	C_i : C_j :		⇒	DEB = min (j,FIN$_i$) FIN = max (j,FIN$_i$)
6)	C_i : C_j :		⇒	DEB = min (FIN$_i$,FIN$_j$) FIN = max (FIN$_i$,FIN$_j$)

.../

FIGURE 4.6 (suite)

7)	C_i :		⇒	DEB = 1 FIN = FIN_i
	C_j :			
8)	C_i :		⇒	DEB = min (DEB_j,FIN_i) FIN = max (DEB_j,FIN_i)
	C_j :			
9)	C_i :		⇒	DEB = DEB_i FIN = j
	C_j :			
10)	C_i :		⇒	DEB = DEB_i FIN = FIN_j
	C_j :			
11)	C_i :		⇒	DEB = 1 FIN = DEB_i
	C_j :			
12)	C_i :		⇒	DEB = min (DEB_i,DEB_j) FIN = max (DEB_i,DEB_j)
	C_j :			

Ainsi, lorsque x_{ij} ($i \neq 1$) est posée égale à 1, C_i et C_j sont réunies, c'est-à-dire elles sont remplacées par une chaîne unique, notée C'_i, dont le début et la fin sont respectivement DEB et FIN. De plus, le poids de C'_i, noté $P(C'_i)$, est égal à $P(C_i) + P(C_j)$ et les sommets sur C_i et C_j sont maintenant considérés comme éléments de C'_i.

La seconde étape de la sous-routine CHCONFIX analyse s'il est possible d'engendrer une contrainte de prévention et, dans l'affirmative, ajoute explicitement celle-ci dans le modèle. Si la variable qui vient d'être fixée à la valeur 1 par la procédure de branchement est incidente au sommet 1, seule la structure de données concernant les chaînes doit être

doit être modifiée; le poids de la chaîne ou du cycle résultant de cette fixation étant le même que celui de la chaîne précédant celle-ci, aucune contrainte de prévention ne peut être calculée et CHCONFIX se termine après la première étape.

On supposera donc dans nos commentaires au sujet de la deuxième étape de CHCONFIX que $i \neq 1$. En posant $x_{ij} = 1$, la chaîne C_i' remplace C_i et C_j. Au début de notre algorithme (rectangle initialisation de la figure 4.4) on s'assure que toute variable $x_{k\ell}$ avec $d_k + d_\ell > D$ est éliminée[1]. De plus, les quantités d_k et D étant engendrées de façon que $d_k < D$ pour $k = 2,\ldots,n$, on pourra toujours fixer une première variable à la valeur 1 sans créer un sous-tour illégal de type 2[2]. Ceci explique pourquoi CHCONFIX est appelée après la procédure de branchement et non avant. Une fois le premier branchement à la valeur 1 effectué, on verra que toute variable $x_{k\ell}$ qui impliquerait la formation d'un sous-tour illégal de type 1 ou de type 2 en prenant la valeur 1 est restreinte à prendre la valeur 0 par la contrainte de prévention engendrée dans CHCONFIX. Il sera donc toujours possible d'effectuer un deuxième branchement à la valeur 1 sans créer immédiatement de sous-tour illégal de type 1 ou de type 2. En général, puisque CHCONFIX est appelée après le $(k-1)^e$ branchement à la valeur 1, le k^e branchement à cette valeur peut toujours être fait sans danger. Il est donc valide de supposer que la variable x_{ij} fixée à la valeur 1 est telle que $P(C_i') < D$, sinon x_{ij} serait forcée de prendre la valeur 0 par la contrainte de prévention précédemment engendrée et ne pourrait être choisie par la procédure de branchement[3].

La forme générale des contraintes de prévention est la suivante:

$$\sum_{(k,\ell)\in R} x_{k\ell} \leq 0^{(4)} \qquad (4.55)$$

---

(1) En pratique on lui impose une borne supérieure nulle.

(2) La première variable fixée au niveau 1 ne peut créer de sous-tour illégal de type 1.

(3) Seules les variables fractionnaires sont considérées par la procédure de branchement; or, dans ce cas, on aura forcément $x_{ij} = 0$.

(4) Puisque $x_{k\ell} \geq 0$ pour tout k et ℓ, on pourrait mettre "=" au lieu de "$\leq$".

où $R \subseteq A$ contient des arêtes satisfaisant aux 3 conditions suivantes:

1. (k,ℓ) incidente à DEB ou FIN et $k \neq 1$;
2. $x_{k\ell} = 1$ implique la formation d'un sous-tour illégal de type 1 ou de type 2;
3. la borne supérieure de $x_{k\ell}$ est non nulle au moment où (4.55) est ajoutée au modèle.

La condition 1 signifie que toute arête qui n'est pas incidente à DEB ou à FIN de même que toute arête incidente au dépôt n'est pas concernée par la formation de C'_i. La condition 3 indique qu'il est inutile d'inclure dans (4.55) une variable déjà fixée à la valeur 0 par la procédure de branchement. La condition 2 réflète simplement le raisonnement sur lequel repose la validité de (4.55). Soit (k,ℓ) satisfaisant à la condition 1 et à la condition 3. Deux cas peuvent se présenter:

1. k = DEB et ℓ = FIN: on a alors que $x_{k\ell} = 1$ implique la formation d'un sous-tour illégal de type 1 et $(k,\ell) \in R$;
2. (k = DEB et $\ell \neq$ FIN) ou bien ($k \neq$ DEB et ℓ = FIN): soit C_p la chaîne réunie à C'_i lorsque $x_{k\ell} = 1$, si $P(C_p) + P(C'_i) > D$ alors $x_{k\ell} = 1$ crée un sous-tour illégal de type 2 et donc $(k,\ell) \in R$.

La seconde étape de CHCONFIX consiste donc à vérifier pour chaque variable $x_{k\ell}$ où (k,ℓ) satisfait aux conditions 1 et 3 si un des deux cas précédents se présente. Une fois R construit, (4.55) est ajoutée au modèle si $R \neq \phi$.

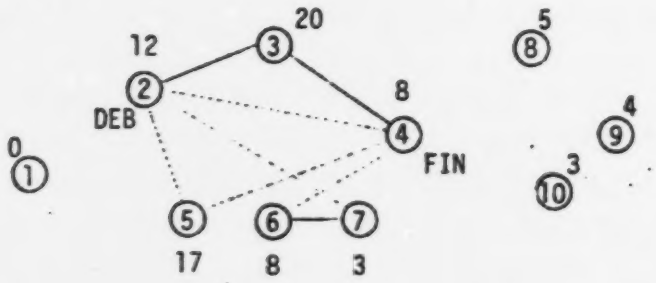
Considérons l'exemple d'un c.m-PVC où n = 10 et D = 45. Supposons que le niveau 6 de l'arbre vient d'être atteint de la façon suivante:

niveau	1	2	3	4	5	6
variable de branchement	$x_{2,6}$	$x_{2,3}$	$x_{4,7}$	$x_{3,9}$	$x_{6,7}$	$x_{3,4} = x_{ij}$
valeur	0	1	0	0	1	1

La variable $x_{ij} = x_{3,4}$ est la variable fixée à 1 juste avant l'appel de CHCONFIX. La figure 4.7 illustre l'état des différentes chaînes formées par la fixation de variables à la valeur 1, la demande de chacune des villes est indiquée à côté du sommet qui lui correspond et les traits pointillés correspondent aux arêtes éléments de R.

FIGURE 4.7

Exemple de la construction de R



Pour l'exemple de la figure 4.7, on obtient R = {(2,4), (2,7), (4,6), (2,5), (4,5)}. La variable $x_{2,4}$ en prenant la valeur 1 causerait la formation d'un sous-tour illégal de type 1; les variables $x_{2,7}$, $x_{4,6}$, $x_{2,5}$ et $x_{4,5}$ réunissant des chaînes dont la demande totale dépasse D, conduiraient à la formation d'un sous-tour illégal de type 2 en se voyant affecter la valeur 1. Il est inutile d'inclure dans R les variables $x_{2,6}$ et $x_{4,7}$ car elles ont été précédemment fixées à la valeur 0 par la procédure de branchement. Après le branchement de $x_{3,4}$ à la valeur 1 la contrainte suivante est donc valide pour le c.m-PVC considéré:

$$x_{2,4} + x_{2,7} + x_{4,6} + x_{2,5} + x_{4,5} \leq 0 \qquad (4.56)$$

Remarquons qu'au niveau 2 de l'arbre la contrainte de prévention (4.57) ci-après sera engendrée:

$$x_{2,5} + x_{3,5} \leq 0 \qquad (4.57)$$

Il n'est donc pas nécessaire d'inclure dans (4.56) la variable $x_{2,5}$.

En utilisant les contraintes de prévention, notre algorithme pour le c.m-PVC prévient, par l'emploi explicite de contraintes, l'occurence de certains sous-tours illégaux. Il diffère donc de l'algorithme de Miliotis [91] pour le PVC et de notre algorithme pour le m-PVC qui ne font qu'éliminer les sous-tours illégaux qui se présentent. Il se rapproche toutefois de l'algorithme de Little et al. [85] pour le PVC où la formation de tout sous-tour illégal est prohibée à l'aide de la procédure de branchement.

Élimination des contraintes devenues invalides ou ineffectives

Lorsque le sous-problème courant est résolu par la méthode du simplex, on compare tout d'abord $\underline{Z}$ à Z^* (voir l'organigramme de la figure 4.4). Si $\underline{Z} \geq Z^*$, ce sous-problème peut être éliminé de nos considérations et un nouveau sous-problème est choisi à un niveau inférieur ou égal dans l'arbre par la sous-routine BACKUP de BB. Certaines variables qui étaient fixées à la valeur 1 peuvent alors devenir libres. La structure de données décrivant les chaînes formées par les variables fixées à la valeur 1 doit alors être modifiée. De plus des contraintes de prévention peuvent devenir invalides et on doit les éliminer du modèle. Il est inutile d'expliquer ici de façon détaillée comment nous avons modifié BB pour effectuer ceci[1]. Disons simplement qu'une fois la sous-routine BAKCUP exécutée, des modifications sont apportées à la structure de données concernant les contraintes du modèle de façon que la sous-routine PURGE, appelée après BACKUP, efface de cette structure les contraintes de prévention devenues invalides. Cette sous-routine PURGE est aussi appelée après le test comparant $\underline{Z}$ à Z^*. Son rôle principal, outre celui mentionné précédemment, est d'éliminer du modèle certaines contraintes devenues ineffectives dans le but d'alléger l'espace-mémoire nécessaire à l'exécution de l'algorithme sur ordinateur. Nous avons modifié la programmation de PURGE pour que:

- les contraintes de degré soient toujours présentes dans le modèle;

(1) Certaines contraintes de prévention devenues invalides peuvent être associées à une rangée de la matrice de base réduite et leur élimination n'est pas simple.

- les coupes de Gomory présentes dans le modèle lorsque le premier branchement est effectué demeurent toujours dans le modèle;
- une contrainte d'élimination d'un sous-tour illégal soit effacée du modèle seulement si elle devient ineffective à un niveau de l'arbre supérieur à celui auquel elle a été ajoutée dans le modèle.

Stratégies de branchement

Nous avons testé dans notre algorithme pour le c.m-PVC deux stratégies de branchement différentes. La première est celle originalement implantée dans BB[1] et elle est décrite dans la section 1.6.1 de ce texte. La deuxième consiste à choisir comme variable de branchement la variable fractionnaire $x_{k\ell}$ qui induit la plus petite augmentation dans l'objectif si elle est fixée à la valeur 1 et de poser $x_{k\ell} = 1$ dans le prochain sous-problème à traiter. La branche de l'arbre où $x_{k\ell} = 0$ sera donc toujours considérée en deuxième lieu. L'idée sous-jacente à cette stratégie de branchement est d'engendrer des contraintes de prévention fortes tôt dans l'arbre et possiblement obtenir ainsi rapidement une bonne solution réalisable pour le c.m-PVC.

4.6.3 Résultats de tests sur ordinateur

Nous avons utilisé pour nos tests sur ordinateur des matrices de coûts symétriques. L'algorithme décrit en 4.6.1 et 4.6.2 peut servir aussi à la résolution d'un c.m-PVC asymétrique en y incorporant des modifications mineures, notamment en ce qui concerne la génération des contraintes de prévention. Les problèmes testés sont aléatoires sauf celui de Dantzig et Ramser [29]. Dans le cas des problèmes aléatoires la matrice

---

(1) Nous l'appellerons la stratégie de branchement de Land et Powell ou bien la première stratégie de branchement.

(2) Nous la dénommerons "notre stratégie de branchement" ou bien la deuxième stratégie de branchement.

des coûts de transport $C = (c_{ij})$ a été engendrée de la façon décrite à la section 1.2. La demande de chaque ville et la capacité de livraison d'un véhicule ont été calculées pour ces problèmes à l'aide de nombres aléatoires b et $b_i \in (0,1)$ provenant d'une distribution uniforme selon (4.58) et (4.59) données ci-après:

$$d_i = [b_i \times 100]^{(1)} \text{ pour } i = 2,\ldots,n \qquad (4.58)$$

$$D = \max_{i=2,\ldots,n} \{d_i\} + \left[b \cdot \left\{ \frac{(\sum_{i=2}^{n} d_i) + 100}{2} \right\} \right] \qquad (4.59)$$

Le choix d'une valeur de D très grande par rapport aux valeurs des d_i conduit à un c.m-PVC facile à résoudre optimalement par notre algorithme car alors le c.m-PVC obtenu est très près d'un m-PVC. De même si la demande de chaque ville est près de la capacité d'un véhicule, les contraintes de prévention engendrées seront très fortes ($|R|$ grand) et la solution optimale du c.m-PVC devrait être obtenue rapidement. Les valeurs de D que nous obtenons par (4.59) sont premièrement telles que $d_i \leq D$ $(i = 2,\ldots,n)$ et deuxièmement, après l'exécution de quelques tests sur ordinateur, nous avons constaté que les c.m-PVC obtenus avec (4.58) et (4.59) sont en général difficiles à résoudre de façon exacte.

Trois paramètres de contrôle binaires conduisent à 8 versions différentes de notre algorithme pour le c.m-PVC. Nous les dénoterons par v_1, v_2 et v_3 où:

$v_1 =$ { 0 : aucune coupe de Gomory n'est engendrée au niveau 0 de l'arbre;
1 : des coupes de Gomory peuvent être ajoutées au modèle au niveau 0 selon la procédure décrite en 4.5.

$v_2 =$ { 0 : les contraintes de prévention ne sont pas utilisées;
1 : sinon

(1) Les valeurs obtenues sont arrondies à l'entier le plus près.

$v_3 = \begin{cases} 0 : \text{la stratégie de branchement implantée dans le code de Land et Powell est utilisée;} \\ 1 : \text{notre stratégie de branchement est utilisée.} \end{cases}$

Les 8 versions de notre algorithme sont définies dans le tableau 4.1:

TABLEAU 4.1

Les 8 versions de notre algorithme pour le c.m-PVC

Version no	v_1	v_2	v_3
1	0	0	0
2	0	1	0
3	1	0	0
4	1	1	0
5	0	0	1
6	0	1	1
7	1	0	1
8	1	1	1

Les versions 1 à 4 utilisent la stratégie de branchement de Land et Powell tandis que les versions 5 à 8 contiennent la stratégie de branchement que nous avons mise au point.

Pour comparer ces 8 versions, chacune d'entre elles a été utilisée pour résoudre 4 problèmes aléatoires de taille n = 15 dans les cas symétrique euclidien et non-euclidien. Les temps d'exécution (sec.) obtenus sont donnés dans le tableau 4.2.

TABLEAU 4.2 / PAGE SUIVANTE

TABLEAU 4.2

Comparaison des 8 versions

CAS NON-EUCLIDIEN

Version/Prob.		1	2	3	4	
	1	211	215	1.0	301	1ere stratégie de branchement
prévention	2	116	231	1.0	160	
coupes	3	132	215	1.0	≥400*	
prévention + coupes	4	75	231	1.1	275	
	5	32	≥400	1.1	228	2e stratégie de branchement
prévention	6	38	346	1.1	224	
coupes	7	85	≥400	1.1	62	
prévention + coupes	8	100	346	1.1	39	

CAS EUCLIDIEN

Version/Prob.		1	2	3	4	
	1	315	56	9.5	70	1ere stratégie de branchement
prévention	2	299	47	9.5	27	
coupes	3	315	32	9.5	72	
prévention + coupes	4	299	28	9.6	72	
	5	≥400	88	40	24	2e stratégie de branchement
prévention	6	≥400	110	43	25	
coupes	7	≥400	124	40	32	
prévention + coupes	8	≥400	73	43	27	

* Temps limite de 400 secondes atteint

D'après le tableau 4.2, il est évident que le c.m-PVC s'avère beaucoup plus difficile à résoudre optimalement que le m-PVC résolu en

un temps moyen de une seconde pour des problèmes de 20 villes [72]. Les 8 versions se départagent peu s'il s'agit d'un c.m-PVC "facile" à résoudre (problème 3 euclidien et non-euclidien). Par contre, dans le cas contraire, les temps d'exécution des 8 versions varient beaucoup. Le facteur prédominant semble la stratégie de branchement utilisée et, pour une même valeur de v_3, les temps d'exécution varient moins. Le tableau 4.2 ne nous permet pas de déterminer de façon définitive laquelle des deux stratégies de branchement est la meilleure.

L'utilisation des contraintes de prévention peut être profitable au niveau de l'énumération mais nécessite du temps d'ordinateur pour les engendrer et les éliminer lorsqu'elles deviennent invalides. Règle générale cependant, elles se sont avérées plus utiles qu'autrement comme en témoigne principalement la version 2 sur les problèmes 1 et 4 non-euclidiens et sur les problèmes 1, 2 et 4 euclidiens. L'emploi des coupes de Gomory souffre du même avantage et du même désavantage[1] que ceux des contraintes de prévention. L'ajout de coupes (voir versions 3 et 7) a diminué le temps d'exécution par rapport aux versions 1 et 5 seulement dans 3 problèmes sur 8 mais par contre semble avoir causé une augmentation dans le temps d'exécution pour 4 autres. Ces augmentations sont toutefois plus petites que les diminutions de temps observées lorsque les coupes se sont avérées utiles.

Les commentaires précédents expliquent pourquoi les versions 4 et 8, utilisant respectivement la stratégie de branchement de Land et Powell et la nôtre avec des coupes de Gomory et des contraintes de prévention, nous sont apparues les plus susceptibles d'être efficaces pour des problèmes de plus grande taille. Nous avons compilé les tableaux suivants où ces deux versions sont utilisées pour résoudre des problèmes aléatoires euclidiens et non-euclidiens de taille entre 15 et 35 villes et aussi celui de Dantzig et Ramser [29] (13 villes). Ces tableaux ne

---

(1) Elles ne sont jamais invalides car on les ajoute dans le modèle au niveau 0 mais il est utile de les éliminer si elles deviennent ineffectives au niveau 0 pour sauver de l'espace-mémoire.

contiennent pas de statistiques concernant le nombre de contraintes d'élimination de sous-tours illégaux et le nombre de contraintes de prévention qui sont engendrées. Il est évident que beaucoup d'entre elles sont ajoutées au modèle vu les temps d'exécution obtenus. Toutefois la sous-routine PURGE en élimine plusieurs au cours du processus et de telles statistiques sont peu utiles. L'utilité de la colonne intitulée "taille maximum de l'inverse" a été expliquée au chapitre 3[(1)]. Le rôle principal de PURGE est de maintenir ce nombre à un niveau acceptable. Lorsqu'on rencontre dans la colonne des temps de résolution "≥ t" cela signifie que le temps limite "t" alloué à ce problème a été atteint.

Considérons tout d'abord le cas des problèmes symétriques non-euclidiens (tableaux 4.3 et 4.4). Encore ici on remarque la très grande difficulté de résolution des c.m-PVC de ce type. Les temps d'exécution varient beaucoup et le problème de plus grande taille résolu contient 30 villes. La stratégie de branchement du code de Land et Powell s'est avérée la plus efficace des 2 stratégies testées lorsque l'optimalité a été atteinte. Toutefois, dans le cas contraire, la qualité de la solution obtenue avec la deuxième stratégie de branchement est supérieure. Celle-ci, en fixant premièrement les variables à la valeur 1, semble trouver une solution réalisable au c.m-PVC plus rapidement, ce qui est clairement un avantage. Même si la solution optimale n'a pas été trouvée, la solution, en main au moment où le temps limite est atteint est fréquemment de coût nettement inférieur à la solution fournie par l'heuristique de Clarke et Wright. A notre avis, C n'étant pas euclidienne, les "savings" ne sont plus nécessairement non-négatifs et cet algorithme se comporte alors beaucoup moins bien. On remarquera d'ailleurs que le coût de la solution optimale de même que le nombre de véhicules utilisés sont habituellement nettement inférieurs aux quantités correspondantes obtenues par l'heuristique de Clarke et Wright. Notons en terminant que le problème de 35 villes testé n'a pas été résolu optimalement mais qu'une solution réalisable de qualité nettement supérieure à celle de la solution heuristique a été trouvée avec la première stratégie de branchement.

---

(1) Voir aussi la section 1.6.1 concernant la matrice inverse de la matrice de base.

TABLEAU 4.3

Problèmes non-euclidiens,
stratégie de branchement de Land et Powell

n	Taille maximum de l'inverse	Nombre d'itérations	Temps de résolution	Coût optimal (nombre de véhicules)	Coût heuristique (nombre de véhicules)
15	27	2103	77.8	315 (4)	369 (4)
	24	69	2.1	244 (2)	294 (2)
	21	45	0.9	229 (3)	407 (6)
	28	11684	≥ 400.0	≤ 447 (5)	486 (6)
	28	5440	139.4	602 (7)	642 (6)
20	26	97	4.1	266 (3)	416 (3)
	36	8633	≥ 400.0	≤ 671 (6)	671 (6)
	34	7505	≥ 400.0	≤ 617 (6)	689 (5)
	23	80	2.6	207 (2)	401 (4)
	22	53	1.2	215 (3)	421 (5)
25	33	988	106.6	199 (2)	345 (4)
	38	7817	≥ 650.0	≤ 544 (5)	560 (5)
	46	9417	≥ 650.0	≤ 1017 (10)	1017 (10)
	34	3361	332.8	244 (2)	369 (4)
	35	6811	≥ 900.0	≤ 457 (5)	457 (2)
	34	1306	135.8	320 (2)	424 (2)
	35	9490	≥ 900.0	≤ 414 (3)	422 (5)
30	39	931	234.1	235 (2)	555 (3)
	36	77	4.6	220 (2)	416 (2)
	40	302	39.4	228 (2)	409 (6)
35	44	12329	≥1800.0	321 (4)	491 (5)

TABLEAU 4.4

Problèmes non-euclidiens, notre stratégie de branchement

n	Taille maximum de l'inverse	Nombre d'itérations	Temps de résolution	Coût optimal (nombre de véhicules)	Coût heuristique (nombre de véhicules)
15	27	2040	68.5	315 (4)	369 (4)
	24	69	2.1	244 (2)	294 (2)
	21	45	0.9	229 (3)	407 (6)
	27	14975	≥ 400.0	≤ 440 (5)	486 (6)
	27	16981	≥ 400.0	≤ 610 (7)	642 (7)
20	26	97	4.1	266 (3)	416 (3)
	34	9537	≥ 400.0	≤ 671 (6)	671 (6)
	33	9069	≥ 400.0	≤ 567 (5)	689 (5)
	25	150	6.7	207 (2)	401 (4)
	22	53	1.2	215 (3)	421 (5)
25	32	5407	418.2	199 (2)	345 (4)
	38	9826	≥ 650.0	≤ 492 (6)	560 (5)
	43	11238	≥ 650.0	≤ 893 (10)	1017 (10)
	33	5707	539.3	244 (2)	369 (4)
	37	12728	≥ 900.0	≤ 457 (5)	457 (5)
	33	6635	613.5	320 (2)	424 (2)
	34	13148	≥ 900.0	≤ 388 (4)	422 (5)
30	41	2908	713.1	235 (2)	555 (3)
	36	77	4.6	220 (2)	416 (2)
	38	3590	402.6	228 (2)	409 (6)
35	46	15377	≥ 1800.0	491 (5)	491 (5)

TABLEAU 4.5

Problèmes euclidiens,
stratégie de branchement de Land et Powell

n	Taille maximum de l'inverse	Nombre d'itérations	Temps de résolution	Coût optimal (nombre de véhicules)	Coût heuristique (nombre de véhicules)
15	31	7967	289.4	626 (4)	654 (4)
	32	328	25.3	403 (2)	405 (2)
	32	854	51.5	492 (4)	492 (4)
	31	13520	≥ 400.0	≤ 679 (4)	679 (4)
	33	3373	94.5	614 (7)	614 (7)
20	35	1110	107.5	412 (3)	440 (3)
	39	7504	≥ 400.0	≤ 1037 (6)	1037 (6)
	37	5925	≥ 400.0	≤ 647 (5)	647 (5)
	32	502	31.4	480 (2)	484 (2)
	32	1323	99.3	520 (3)	561 (3)
25	43	3949	≥ 650.0	≤ 545 (2)	559 (2)
	44	4913	≥ 650.0	≤ 664 (4)	664 (4)
	44	8512	≥ 650.0	≤ 1148 (10)	1148 (10)
	37	483	38.1	497 (2)	523 (2)
	42	9396	≥ 900.0	≤ 567 (4)	567 (4)
	39	710	70.6	562 (2)	622 (2)
	38	8875	≥ 900.0	≤ 642 (4)	642 (4)
30	44	5096	≥ 1000.0	584 (2)	587 (2)
	48	6994	≥ 1000.0	776 (6)	776 (6)
	48	8580	1420.6	503 (2)	584 (2)
	41	150	11.3	518 (2)	572 (2)
13*	29	8976	310.5	290 (4)	290 (4)

(*) Problème de Dantzig et Ramser [29].

TABLEAU 4.6

Problèmes euclidiens, notre stratégie de branchement

n	Taille maximum de l'inverse	Nombre d'itérations	Temps de résolution	Coût optimal (nombre de véhicules)	Coût heuristique (nombre de véhicules)
15	30	5043	153.1	626 (4)	654 (4)
	32	301	23.4	403 (2)	405 (2)
	32	1917	102.1	492 (4)	492 (4)
	30	14200	≥ 400.0	≤ 679 (4)	679 (4)
	30	17425	> 400.0	614 (7)	614 (7)
20	35	1632	145.0	412 (3)	440 (3)
	35	7997	≥ 400.0	≤ 1037 (6)	1037 (6)
	44	7387	≥ 400.0	≤ 647 (5)	647 (5)
	33	1169	82.4	480 (2)	484 (2)
	33	2236	142.2	520 (3)	561 (3)
25	44	4968	≥ 650.0	≤ 537 (2)	559 (2)
	37	7346	≥ 650.0	≤ 664 (4)	664 (4)
	40	6850	≥ 650.0	≤ 1148 (10)	1148 (10)
	37	5899	406.3	497 (2)	523 (2)
	38	12553	≥ 900.0	≤ 567 (4)	567 (4)
	37	10558	≥ 900.0	≤ 615 (2)	622 (2)
	40	10628	≥ 900.0	≤ 642 (4)	642 (4)
30	43	5636	> 900.0	≤ 585 (2)	587 (2)
	47	7099	≥ 900.0	≤ 776 (6)	776 (6)
	45	14087	≥ 1800.0	≤ 549 (2)	584 (2)
	43	753	83.5	518 (2)	572 (2)
13*	29	57993	≥ 1800.0	≤ 290 (4)	290 (4)

(*) Problème de Dantzig et Ramser [29].

Analysons maintenant les résultats des tableaux 4.5 et 4.6 concernant les problèmes symétriques euclidiens. Que l'optimalité soit atteinte ou non, la première stratégie de branchement est supérieure. Le problème de plus grande taille résolu optimalement contient 30 villes. La solution heuristique de Clarke et Wright semble en général de meilleure qualité que dans le cas non-euclidien bien que pour les 2 problèmes de 30 villes résolus la solution optimale est de coût nettement inférieur. Le problème de Dantzig et Ramser a été résolu en 310 secondes par la première stratégie de branchement. Soulignons que ce problème a été résolu par la version 1[(1)] en 819 secondes, ce qui illustre l'utilité des contraintes de prévention, surtout dans les problèmes serrés.

Nos algorithmes pour le m-PVC et le c.m-PVC sont basés en général sur le même processus de relaxation et réintroduction des contraintes d'élimination de sous-tours illégaux. Comme en témoignent les résultats sur ordinateur précédents, une telle méthode fournit des résultats moins spectaculaires pour le c.m-PVC que sur le m-PVC. A notre avis, l'explication de ce phénomène est la suivante. Soit $V(P_D)$ la valeur optimale du problème contenant uniquement les contraintes de degré. Soit V(m-PVC) et V(c.m-PVC) les valeurs optimales du, respectivement, m-PVC et c.m-PVC associés à P_D. Les résultats de nos chapitres 3 et 4 indiquent clairement que, en moyenne:

$$V(P_D) < V(\text{m-PVC}) \ll V(\text{c.m-PVC}) \qquad (4.60)$$

On suppose évidemment dans (4.60) que D n'est pas trop grand[(2)]. Notre méthode pour le m-PVC et le c.m-PVC débute par la solution de P_D. On a vu au chapitre 3 qu'en général $V(P_D)$ est très près de V(m-PVC) et que l'ajout d'un très petit nombre de contraintes d'élimination de sous-tours illégaux et de coupes de Gomory suffit à obtenir V(m-PVC) en partant de $V(P_D)$ c'est-à-dire à combler la différence V(m-PVC) - $V(P_D)$. Pour solutionner le c.m-PVC, il faut en plus, par l'insertion de contraintes du même type, passer de V(m-PVC) à V(c.m-PVC) en comblant une marge

(1) Sans les coupes et les contraintes de prévention.

(2) Sinon le c.m-PVC est à peu près équivalent au m-PVC.

beaucoup plus grande. Il s'agit là de la première raison qui rend le c.m-PVC plus difficile à résoudre que le m-PVC même si l'utilisation des contraintes de prévention accélère habituellement le processus. Une deuxième raison réside dans la nature même des contraintes de capacité qui sont "opposées" à l'objectif. Ce dernier conduit souvent à des tours contenant plusieurs villes et de tels tours sont susceptibles de dépasser la capacité d'un véhicule. Ce n'est donc que graduellement et lentement qu'une solution est amenée à respecter les contraintes de capacité. Si le nombre de contraintes de prévention et d'élimination de sous-tours illégaux présentes est insuffisant, le simplex réussit fréquemment à trouver une solution qui, tout en satisfaisant les contraintes actuelles, contient encore un "grand tour" qu'il faudra "briser" par une nouvelle contrainte, etc. Le temps pris pour combler V(c.m-PVC) - V(m-PVC) est donc souvent grand et a fortiori les temps d'exécution obtenus sont plus grands que pour le m-PVC.

4.6.4 Comparaison avec les autres méthodes pour le c.m-PVC

Nous avons relevé dans la littérature très peu de données nous permettant de comparer notre algorithme aux autres méthodes existantes résolvant le c.m-PVC optimalement et ceci pour 2 raisons principales:

1. en majorité ces méthodes sont heuristiques;
2. le problème considéré n'est pas identique au c.m-PVC (m fixé, contraintes autres que la contrainte capacité, etc.);
3. des tests sur des problèmes asymétriques ont été effectués.

Christofides [16] mentionne qu'un problème de type PEV contenant 31 villes a été résolu optimalement. Dans cet article Christofides ne décrit pas de façon précise les contraintes exactes dont on a tenu compte. On sait toutefois que les contraintes de capacité ainsi que plusieurs autres y ont été considérées.

Christofides, Mingozzi et Toth [21] ont mis au point trois méthodes exactes de résolution d'un problème similaire au c.m-PVC mais où:

1. m est fixé à l'avance[1];
2. les contraintes de temps maximum (4.47) sont imposées sur les tours.

Ces trois méthodes sont décrites brièvement dans la section 4.5.2 de ce texte. Elles ont été testées sur 10 problèmes asymétriques de taille variant entre 10 et 25 villes. Une seule de ces méthodes a pu résoudre optimalement les 10 problèmes en un temps inférieur ou égal à 250 secondes (CDC 7600). Les temps d'exécution obtenus démontrent eux aussi une grande variance.

Notons en dernier lieu que la programmation dynamique avec relaxation d'états [22] a été utilisée récemment pour calculer une borne inférieure sur le c.m-PVC où m est fixé. Pour 10 problèmes de 10 à 25 villes, la borne obtenue s'est avérée être au minimum à 93% du coût optimal.

Il nous est clairement difficile de comparer de façon détaillée l'efficacité de notre méthode à celle des méthodes des deux articles précédents surtout en raison du fait que le problème considéré n'est pas le même. A notre avis toutefois, les algorithmes de Christofides et al. [21] et la méthode de Christofides [16], basée sur la génération des tours réalisables, s'avèreront surtout efficaces pour des problèmes où la restriction de capacité est forte; par contre, la méthode que nous avons développée, basée sur l'élimination de solutions non réalisables, sera surtout performante pour des problèmes où la contrainte de capacité est moins restrictive. Une constance se dégage cependant de nos travaux et de ceux des auteurs précédemment cités: l'ajout de contraintes de capacité au m-PVC le rend extrêmement difficile à résoudre de façon optimale et les tailles maximales atteintes sont du même ordre de grandeur: 25 à 30 villes.

4.7 Conclusion

Nos résultats pour le c.m-PVC peuvent se résumer ainsi:

---

(1) A notre avis, il serait difficile de modifier ces méthodes pour traiter du cas où m est libre (voir 3.6 à propos des méthodes basées sur les m-arbres ou m-arborescences).

1. nous avons construit un nouveau modèle pour ce problème, généralisant celui de Dantzig et al. [28] pour le PVC au cas du c.m-PVC;

2. nous avons prouvé que notre modèle pour le c.m-PVC se déduit de celui de Gavish et Graves [45] pour le même problème à l'aide de la technique de décomposition de Benders [9];

3. nous avons construit un nouvel algorithme pour le c.m-PVC qui s'avère compétitif en temps et en taille des problèmes résolus par rapport aux autres méthodes exactes résolvant des problèmes similaires(1) au c.m-PVC;

4. notre algorithme pour le c.m-PVC est très flexible et peut s'adapter facilement aux cas où m est fixé, où des coûts fixes sur les commis sont considérés et où C est asymétrique.

Clairement il reste beaucoup à faire pour augmenter la taille des c.m-PVC que l'on peut traiter de façon exacte. Une suite possible de nos travaux serait de tenter de résoudre notre modèle en utilisant une approche du type "relaxation lagrangienne restreinte" (voir les travaux de Balas et Christofides [2] sur le PVC) où un sous-ensemble des contraintes d'élimination de sous-tours illégaux sont insérées dans l'objectif à l'aide de multiplicateurs de Lagrange. La borne inférieure ainsi obtenue pourrait être utilisée dans une méthode de "Branch-and-Bound" et la valeur des multiplicateurs serait calculée par une technique de sous-gradients [109].

---

(1) m fixé, contraintes de temps maximum imposées, cas asymétrique, etc.

CHAPITRE 5

LE PROBLEME DES m VOYAGEURS DE COMMERCE AVEC p DEPOTS ((p,m)-PVC)

5. LE PROBLEME DES m VOYAGEURS DE COMMERCE AVEC p DEPOTS ((p,m)-PVC)

5.1 Introduction

Les travaux décrits dans les chapitres précédents indiquent clairement qu'une technique de relaxation - réintroduction de contraintes sur les sous-tours illégaux utilisée conjointement avec une méthode d'hyperplans de coupe ou une méthode de "Branch-and-Bound" peut s'avérer très efficace pour résoudre optimalement plusieurs problèmes de type PEV[1]. Nous avons donc décidé d'utiliser un tel processus dans le cas d'un m-PVC où les tours pourraient être associés à plus d'un dépôt. Ce problème se retrouve dans plusieurs situations pratiques [120,126] et, à notre connaissance, aucun algorithme exact n'a été développé pour le résoudre. Il sera dénoté par (p,m)-PVC avec p le nombre de dépôts et m le nombre de tours à construire. La forme de ceux-ci est assujettie aux mêmes contraintes que dans le m-PVC avec la restriction qu'un tour contienne exactement un dépôt. La localisation des p dépôts parmi les n villes du problème est connue à l'avance et au moins un tour devra débuter et se terminer à chaque dépôt. Toute solution réalisable d'un (p,m)-PVC contiendra donc au moins p tours. Par ailleurs m peut être fixe ou variable.

Deux cas particuliers importants du (p,m)-PVC ont été analysés de façon intensive dans la littérature. Il s'agit premièrement du PVC, selon la notation utilisée ici un (1,1)-PVC, qui a été considéré au chapitre 2 et deuxièmement du m-PVC, notamment un (1,m)-PVC, qui a été étudié au chapitre 3.

Nous avons retrouvé dans la littérature très peu d'articles concernant le (p,m)-PVC. Rao [102], dans un article récent contenant une transformation du m-PVC symétrique en un PVC du même type (section 3.2.6), montre qu'un (2,2)-PVC asymétrique peut aussi être transformé en un PVC. Rao mentionne de plus que l'équivalence entre un (p,m)-PVC et un PVC semble en général difficile à établir.

---

(1) Problème d'expédition de véhicules.

Dans le rapport de Bodin et al. [12] contenant un résumé des connaissances actuelles au sujet des problèmes de type PEV, on retrouve les articles [48,52,121] concernant le (p,m)-PVC où des contraintes de capacité (voir chapitre 4) sont aussi incluses dans le problème. D'autres références connexes au (p,m)-PVC où les dépôts ne sont pas localisés a priori sont données dans le chapitre 6. L'ajout de contraintes de capacité dans le contexte du (p,m)-PVC et/ou la localisation des dépôts n'ont pas été considérés ici.

Pour imposer la présence d'un seul dépôt par tour, nous avons dû mettre au point, en plus des contraintes sur les sous-tours illégaux, des contraintes d'élimination de "chaînes"(1) pour parvenir à formuler mathématiquement le (p,m)-PVC. En pratique, dans certaines situations, le tour d'un commis peut contenir plus d'un dépôt. Le problème considéré ici ne permet pas cette liberté et il est donc plus difficile à résoudre, à tout le moins pour notre algorithme.

5.2 Formulation du (p,m)-PVC

Comme précédemment $N = \{1,\ldots,n\}$ dénote l'ensemble des villes du (p,m)-PVC. L'ensemble K ($\subseteq N$) dénotera l'ensemble des dépôts. La matrice $C = (c_{ij})$ est encore ici la matrice des coûts de transport inter-villes. Puisque le tour d'un commis ne doit contenir qu'un seul dépôt, on aura $c_{ij} = \infty$ si i et j font partie de l'ensemble des dépôts et donc les variables x_{ij} avec i et $j \in K$ ne sont pas définies.

On suppose dans ce chapitre que C est symétrique(2). L'extension de notre formulation du (p,m)-PVC au cas asymétrique s'effectue simplement. Toutefois au niveau algorithmique, le cas asymétrique

---

(1) De façon plus précise, contraintes d'élimination de chaînes "entre deux dépôts".

(2) Le problème considéré ici est en fait le (p,m)-PVC symétrique.

semble en général plus difficile à résoudre lorsqu'une approche comme la nôtre est utilisée car le nombre de variables est alors approximativement doublé. Le chapitre 3 contient une discussion sur ce sujet.

Nous distinguerons deux types de (p,m)-PVC: ceux euclidiens et ceux non-euclidiens selon que C est respectivement euclidienne ou non-euclidienne. Cette propriété de C influence le type de solutions optimales obtenues: lorsque C est complète et satisfait l'inégalité du triangle[1], il est facile de se rendre compte que le nombre optimal de tours par dépôt est 1.

Dans la formulation du (p,m)-PVC qui suit, on suppose que m_k ($k \in K$) représente le nombre de commis associés au dépôt k. En général m_k peut être fixe ou variable et dans ce dernier cas il existe plusieurs façons de borner inférieurement ou supérieurement m_k. Quelques-unes d'entre elles seront données plus loin dans le texte. Nous supposerons ici que $1 \leq m_k \leq \bar{m}_k$ où $\bar{m}_k$ est une constante prédéterminée. La lettre "m" dans le notation (p,m)-PVC représente le nombre total de commis utilisés, c'est-à-dire $m = \sum_k m_k$.

Les tours des commis dans un (p,m)-PVC doivent satisfaire aux conditions suivantes:

1. chaque ville autre qu'un dépôt est visitée exactement une fois par un seul commis;
2. chaque ville-dépôt k est visitée m_k fois;
3. le tour d'un commis contient exactement un dépôt.

La formulation du (p,m)-PVC présentée ici est similaire à celle que nous avons mise au point pour le m-PVC. Seules les contraintes d'élimination de chaînes sont vraiment nouvelles et leur écriture nécessi-

---

(1) $c_{ik} + c_{kj} \geq c_{ij}$ pour tout $i,j,k \in N$.

te la notation suivante:

- $(i_1,\ldots,i_{|T|})$ une permutation des éléments de $T \subseteq N$;
- $\sigma(T)$ l'ensemble des permutations de T;
- $x'_{ij} = \begin{cases} x_{ij} \text{ si } i < j \\ x_{ji} \text{ sinon} \end{cases}$

A toute permutation $(i_1,\ldots,i_{|T|})$ dans $\sigma(T)$, on peut associer une chaîne u sur les sommets de T, notamment la chaîne associée à la séquence:

$$((k_1,\ell_1),\ldots,(k_{|T|-1},\ell_{|T|-1})) \text{ où } (k_j,\ell_j) = \begin{cases} (i_j,i_{j+1}) \text{ si } i_j < i_{j+1} \\ (i_{j+1},i_j) \text{ sinon} \end{cases}$$

L'expression suivante:

$$\sum_{t=1}^{|T|-1} \; \sum_{i_t,i_{t+1} \in T} x'_{i_t,i_{t+1}}$$

représente la somme des variables associées aux arêtes de u.

Le modèle suivant est valide pour le (p,m)-PVC:

(P)

$$\text{Min } Z = \sum_{i<j} c_{ij}\, x_{ij} \qquad (5.1)$$

<u>assujettie à</u>:

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2m_k \qquad (k \in K) \qquad (5.2)$$

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2 \qquad (k \in N-K) \qquad (5.3)$$

$$\sum_{\substack{i \in S, j \in S \\ i<j}} x_{ij} \leq |S|-1 \qquad (S \subset N,\ S \cap K = \phi) \qquad (5.4)$$

$$\sum_{t=1}^{|T|-1} \sum_{i_t, i_{t+1} \in T} x'_{i_t, i_{t+1}} \leq |T|-2 \quad \text{(pour toute permutation dans } \sigma(T), T \subseteq N, |T| \geq 3, |T \cap K| = 2) \tag{5.5}$$

$$x_{ij} = 0,1 \quad (i,j \in N-K, i < j) \tag{5.6}$$

$$x_{ij} = 0,1,2 \quad (i \in K \text{ et } j \in N-K \text{ ou bien } i \in N-K \text{ et } j \in K;\ i < j) \tag{5.7}$$

$$1 \leq m_k \leq \bar{m}_k \quad (k \in K) \tag{5.8}$$

(1)

Les variables x_{ij} du modèle P s'interprètent de la façon usuelle. Encore ici on permet un aller-retour d'un dépôt à une ville par (5.7). Notons qu'une dépendance du coût de transport inter-villes vis-à-vis le commis utilisé ou encore selon le dépôt à partir duquel origine le commis augmenterait le nombre de variables dans P et le rendrait probablement plus difficile à résoudre.

L'objectif (5.1) évalue le coût total des tours construits. Pour modéliser le cas où un coût fixe f_k est encouru lors de l'utilisation d'un commis au dépôt k, le terme linéaire $\sum_{k \in K} f_k m_k$ doit être ajouté à (5.1). D'après notre expérience, de tels coûts n'ajoutent rien à la difficulté de résolution d'un problème et n'ont pas été considérés dans nos tests sur ordinateur.

Les contraintes (5.2) et (5.3) de P constituent les contraintes de degré pour le (p,m)-PVC: chaque visite à un sommet augmente de 2 unités son degré. Ainsi un sommet-dépôt, visité m_k fois, aura un degré de $2m_k$ et un sommet associé à une ville non dépôt aura un degré de 2, celle-ci étant visitée exactement une fois. Tel que mentionné précédemment, nous aurions pu borner les m_k d'une façon différente dans le cas où ils sont libres(2). Les contraintes (5.9) et (5.10) ci-après illustrent 2 autres possibilités:

---

(1) L'intégrité sur les variables m_k est assurée par (5.2), (5.6) et (5.7).

(2) On peut même à la rigueur en fixer certains et en laisser d'autres libres.

$$\underline{m} \leq \sum_{k \in K} m_k \leq \bar{m} \tag{5.9}$$

$$\sum_{k \in K} m_k = \bar{m} \tag{5.10}$$

avec $\underline{m}$ et $\bar{m}$, respectivement, une borne inférieure et supérieure sur le nombre total de commis. Remarquons qu'en utilisant (5.9) ou (5.10) au lieu de (5.8), on permet une solution où $m_k = 0$ pour un certain $k \in K$[1]. Un tel dépôt serait dit alors "fermé" et un certain aspect "localisation" serait inséré dans le (p,m)-PVC. Il est clair qu'on pourrait aussi dans (5.8) utiliser une borne inférieure sur m_k différente de 1. Toutes ces possibilités se traitent avec une égale facilité par notre algorithme.

Les contraintes (5.4) et (5.5) servent à empêcher la formation de sous-tours illégaux. Considérons une solution entière de P satisfaisant à (5.2), (5.3) et à (5.6) jusqu'à (5.8). Si cette solution n'est pas légale pour le (p,m)-PVC, forcément elle contient au moins un des deux types suivants de sous-tours, sinon les deux:

1. un sous-tour constitué de villes dans N-K, c'est-à-dire un sous-tour ne contenant aucun dépôt; on appellera un tel sous-tour, un sous-tour illégal de type 1;
2. un sous-tour constitué de villes dont au moins deux sont dans K, c'est-à-dire un sous-tour contenant plus d'un dépôt, on appellera un tel sous-tour, un sous-tour illégal de type 2.

Les contraintes (5.4) servent à empêcher la formation de sous-tours illégaux de type 1. Elles opèrent de façon identique à (2.24) ou (3.27) qui ont été étudiées aux chapitres 2 et 3 de ce texte et ne seront pas explicitées plus en détails.

Les contraintes (5.5) empêchent l'apparition de sous-tours illégaux de type 2. Elles sont nécessaires dans le cas d'un sous-tour con-

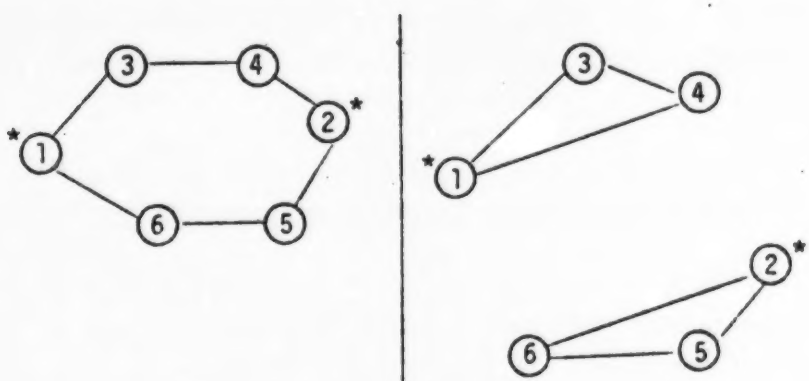
---

(1) Dans le cas où certaines variables m_k peuvent prendre la valeur 0, une formulation plus simple du (p,m)-PVC peut être mise au point.

tenant 2 dépôts ou plus car les contraintes de (5.4) ne sont pas valides pour ce type de sous-tour tel qu'en témoigne le contre-exemple de la figure 5.1 où les sommets 1 et 2 constituent 2 dépôts.

FIGURE 5.1

Illustration de la non validité de (5.4)
pour les sous-tours de plus d'un dépôt



$S = \{1,\ldots,6\}$

$$\sum_{i\in S, j\in S, i<j} x_{ij} = 6 = |S| > |S|-1$$

sous-tour illégal

$S = \{1,\ldots,6\}$

$$\sum_{i\in S, j\in S, i<j} x_{ij} = 6 = |S| > |S|-1$$

2 sous-tours légaux

Comme on le constate par la partie droite de la figure 5.1, les contraintes de (5.4) sont trop fortes dans le cas d'un ensemble S contenant plus d'un dépôt car elles peuvent empêcher la formation de solutions légales pour le (p,m)-PVC.

Les contraintes d'élimination de chaînes sont basées sur le raisonnement suivant. Considérons un sous-tour contenant au moins deux dépôts notés k et ℓ. Il existe forcément une chaîne reliant k et ℓ. Soit T l'ensemble des sommets de cette chaîne et $(i_1,\ldots,i_{|T|})$ la permutation dans $\sigma(T)$ associée à cette chaîne. Il s'ensuivra alors que pour cette solution:

$$\sum_{\substack{t=1 \\ i_t, i_{t+1} \in T}}^{|T|-1} x^1_{i_t, i_{t+1}} = |T|-1 > |T|-2$$

donc une des contraintes de (5.5) est violée par cette solution. En fait, (5.5) empêche la formation de toute chaîne entre 2 dépôts et ainsi aucun sous-tour contenant 2 dépôts ou plus peut se former.

Il est facile de montrer que le modèle P donné précédemment se réduit au modèle que nous avons construit pour le m-PVC symétrique lorsque p = 1.

5.3 Une méthode exacte de résolution du (p,m)-PVC

Le modèle du (p,m)-PVC décrit en 5.2 contient un nombre extrêmement grand de contraintes même pour n relativement petit: comme on l'a vu dans les chapitres précédents, la présence de contraintes du type de (5.4) suffit à rendre un modèle impossible à résoudre directement; dans le cas du (p,m)-PVC, en plus des contraintes de (5.4), il faut compter les contraintes d'élimination de chaînes (5.5) qui sont elles aussi en très grand nombre. L'algorithme que nous avons construit pour le (p,m)-PVC est similaire à notre algorithme NORMAL pour le m-PVC et sera décrit brièvement:

Etape 1:

Une relaxation continue du (p,m)-PVC contenant les contraintes (5.2), (5.3) et les bornes sur les variables induites par (5.6) jusqu'à (5.8) est tout d'abord résolue.

Etape 2:

Une sous-routine vérifie si la solution courante est entière. Si oui, l'étape 3 est exécutée. Si au contraire la solution n'est pas entière, cette sous-routine ajoute une coupe de Gomory au modèle et

le problème est réoptimisé. L'étape 2 est répétée jusqu'à ce que la solution obtenue soit entière. Notons qu'après chaque réoptimisation, les coupes de Gomory ineffectives sont éliminées de façon à économiser de l'espace-mémoire. La sous-routine PURGE décrite en 1.6.1 sert à cette fin.

Etape 3:

Une sous-routine vérifie si la solution contient des sous-tours illégaux. Si oui, des contraintes de (5.4) et/ou (5.5) sont adjointes au modèle, une phase de réoptimisation suit et on va à l'étape 2. Sinon, la solution obtenue est optimale et on termine.

Nous voudrions poser les commentaires suivants au sujet de cet algorithme. Tout d'abord une méthode de coupes a été préférée à une méthode de "Branch-and-Bound" pour restaurer l'intégrité. Nous avons déjà discuté des avantages et des désavantages de chacun de ces deux types de méthodes. Disons simplement qu'après quelques tests sur ordinateur, les coupes de Gomory s'étant bien comportées, nous avons décidé de les utiliser pour tous nos tests. La sous-routine INTCON de notre algorithme pour le m-PVC a été utilisée pour les engendrer.

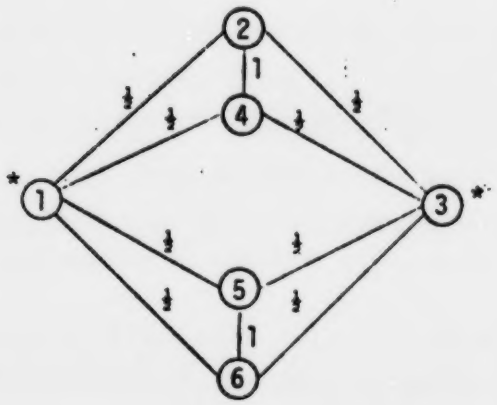
Soulignons qu'il est préférable et même parfois nécessaire dans le cas du (p,m)-PVC d'atteindre l'intégrité avant d'engendrer des contraintes de (5.5). Pour expliquer la raison de ce phénomène, supposons que par un certain processus d'étiquetage[1], la composante connexe illustrée dans la figure 5.2 a été identifiée.

Supposons que les sommets-dépôts, identifiés par une étoile dans la figure 5.2, sont les sommets 1 et 3 avec $m_1 = m_3 = 1$. Soit $S = \{1,2,3,4,5,6\}$ et $T = \{1,2,3\}$. Considérons la chaîne $((1,2),(2,3))$ entre les 2 dépôts. La somme des variables associées aux arêtes de cette chaîne, notamment, $x_{1,2}+x_{2,3}$, est égale à 1, ce qui est plus petit ou égal à $|T|-2$. La contrainte de (5.5) associée à cette chaîne ne briserait donc pas celle-ci, ce qui ne se produirait pas si la so-

(1) Ce type de processus a été décrit au chapitre 3 dans le cas du m-PVC.

FIGURE 5.2

Exemple d'une composante connexe
dans une solution fractionnaire



lution était entière. Notons qu'en supposant que S ne contient aucun dépôt, la contrainte de (5.4) associée à S est violée par cette solution fractionnaire car:

$$\sum_{\substack{i\in S, j\in S \\ i<j}} x_{ij} = 4 = |S| > |S|-1$$

donc (5.4) pourrait être utilisée pour éliminer un tel "sous-tour illégal". Cette dernière remarque explique d'ailleurs pourquoi nous avons pu construire l'algorithme RENVERSE pour le m-PVC.

En résumé donc, les contraintes de (5.5) doivent être engendrées dans certains cas à partir d'une solution entière tandis que (5.4) peut rendre irréalisable une solution qu'elle soit entière ou non[1]. Nous avons décidé d'atteindre une solution entière avant d'engendrer des contraintes (5.5) et/ou (5.4) bien que ces dernières contraintes peuvent

(1) La somme des variables d'une composante connexe est constante que la solution soit entière ou non, ce qui n'est pas le cas de la somme des variables sur une chaîne.

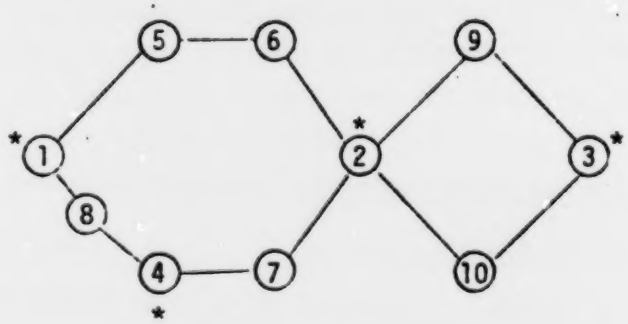
être ajoutées au modèle à partir d'une solution fractionnaire et ceci dans le but de minimiser le nombre de fois que le processus d'étiquetage devra être exécuté.

La génération des contraintes de (5.4) a été expliquée au chapitre 3 car ces contraintes sont identiques à (3.27). Une contrainte de (5.4) est ajoutée au problème courant pour chaque sous-tour de type 1 que sa solution contient.

La génération des contraintes de (5.5) a été effectuée de la façon suivante. Chaque composante connexe est premièrement divisée en sous-tours dont tous les sommets ont un degré égal à 2. On appelle ceux-ci des sous-tours simples. Deuxièmement, pour chaque sous-tour simple avec $S \subset N$ comme ensemble de sommets, $|S \cap K|$ contraintes de (5.5) sont engendrées, chaque sous-tour simple étant constitué de $|S \cap K|$ chaînes reliant 2 dépôts. Considérons la composante connexe illustrée dans la figure 5.3 où les dépôts sont les sommets 1,2,3 et 4:

FIGURE 5.3

Exemple d'une composante connexe avec 4 dépôts



Cette composante connexe peut être divisée en 2 sous-tours simples: le premier sur l'ensemble de sommets $S_1 = \{1,5,6,2,7,4,8\}$ et le deuxième sur $S_2 = \{2,9,3,10\}$. A partir du sous-tour sur S_1, les 3 contraintes de chaîne suivantes sont engendrées:

Contraintes d'élimination de chaîne #1:

$$T = \{1,5,6,2\},\ T \cap K = \{1,2\}$$

$$x_{1,5} + x_{5,6} + x_{2,6} \leq 2 \qquad (5.11)$$

Contraintes d'élimination de chaîne #2:

$$T = \{2,7,4\},\ T \cap K = \{2,4\}$$

$$x_{2,7} + x_{4,7} \leq 1 \qquad (5.12)$$

Contraintes d'élimination de chaîne #3:

$$T = \{4,8,1\},\ T \cap K = \{4,1\}$$

$$x_{4,8} + x_{1,8} \leq 1 \qquad (5.13)$$

Chacune des contraintes précédentes est violée par une solution contenant la composante connexe de la figure 5.3. A partir du sous-tour sur S_2, 2 contraintes de (5.5) sont engendrées, notamment:

Contraintes d'élimination de chaîne #4:

$$T = \{2,9,3\},\ T \cap K = \{2,3\}$$

$$x_{2,9} + x_{3,9} \leq 1 \qquad (5.14)$$

Contraintes d'élimination de chaîne #5:

$$T = \{3,10,2\},\ T \cap K = \{3,2\}$$

$$x_{3,10} + x_{2,10} \leq 1 \qquad (5.15)$$

Remarquons que le fait d'imposer $|S \cap K|$ contraintes différentes à partir d'un sous-tour simple sur S est plus fort que la génération d'une seule contrainte du type:

$$\sum_{\substack{i \in S, j \in S \\ i<j \\ x_{ij}=1}} x_{ij} \leq |S| - |S \cap K| \qquad (5.16)$$

et a fortiori plus fort que:

$$\sum_{\substack{i\in S, j\in S \\ i<j \\ x_{ij}=1}} x_{ij} \leq |S| - 1 \qquad (5.17)$$

qui toutes deux auraient pour effet aussi de briser un sous-tour illégal de type 2.

Dans notre exemple (5.16) s'écrit si $S = S_1$:

$$x_{1,5} + x_{5,6} + x_{2,6} + x_{2,7} + x_{4,7} + x_{4,8} + x_{1,8} \leq 4 \qquad (5.18)$$

La contrainte (5.18) est la somme des contraintes (5.11), (5.12) et (5.13).

5.4 Tests de performance

Des (p,m)-PVC aléatoires de taille variant entre 20 et 60 villes ont été testés et résolus pour différentes valeurs de p dans le cas symétrique euclidien et non-euclidien. Pour tous les problèmes, la valeur de $\bar{m}_k$ a été fixée à n/5. Les détails techniques concernant la génération des problèmes-tests sont donnés dans le chapitre 1. Le code utilisé a été construit à partir de la version en arithmétique entière écrite par Miliotis [90], du programme MIF (section 1.6) de Land et Powell [71]. Les limites suivantes ont été imposées lors de la résolution des problèmes-tests:

- temps limite : 250 secondes (dénoté par TL);
- nombre total de contraintes : 130
- taille maximum de l'inverse : 110
- nombre maximum de coefficients dans la matrice des contraintes : 15000
- valeur maximum du déterminant : 10^6

Les résultats de nos tests de performance sont donnés dans les tableaux 5.1 et 5.2. Le (p,m)-PVC est en général beaucoup plus difficile à résoudre que le m-PVC. De plus, pour des problèmes avec le même nombre de villes, la variance dans les temps de résolution obtenus pour le (p,m)-PVC est beaucoup plus grande que pour le m-PVC. Pour certains problèmes, très peu de sous-tours de type 2 furent découverts et notre algorithme s'est bien comporté. La valeur de p dans de tels problèmes est souvent petite. Pour d'autres problèmes cependant, un grand nombre de contraintes d'élimination de chaînes est engendré. Il s'ensuit l'ajout dans le modèle de plusieurs coupes de Gomory et l'algorithme peut alors se terminer soit par manque d'espace-mémoire, soit à cause d'un déterminant trop grand ou par un temps de résolution élevé si la solution optimale a été atteinte. Ainsi le fait de permettre la présence de plus d'un dépôt par sous-tour, éliminant la nécessité des contraintes d'élimination de chaînes, rendrait le (p,m)-PVC beaucoup plus facile à résoudre.

5.5 Conclusion

La méthode que nous avons construite pour résoudre le (p,m)-PVC constitue, à notre connaissance, le premier algorithme optimal pour ce problème. Il a été traité auparavant de façon heuristique par [52,121] avec (5.9) au lieu de (5.8). Les tailles atteintes sont respectables malgré la difficulté de résolution du (p,m)-PVC et on peut conclure que notre relaxation a relativement bien fonctionné.

Un nouveau type de contraintes a été introduit dans ce chapitre, notamment les contraintes d'élimination de chaînes. Elles sont moins "fortes" que les contraintes de Dantzig et al. (5.4) car elles concernent habituellement un ensemble de variables plus restreint. Elles sont toutefois très flexibles et peuvent être employées dans plusieurs problèmes de type PEV.

TABLEAU 5.1

Problèmes non-euclidiens

n = nombre de villes	p = nombre de dépôts	nombre de coupes de Gomory engendrées	nombre de sous-tours de type 1 rencontrés	nombre de sous-tours de type 2 rencontrés	nombre de contraintes de (5.5) engendrées	taille maximum de l'inverse	nombre d'itérations du simplex	temps de résolution	diagnostic
20	2	1	12	1	2	29	38	2.10	√
20	2	0	12	0	0	25	37	1.72	√
20	3	57	7	6	16	37	313	47.67	√
20	3	2	6	0	0	27	32	1.70	√
20	5	9	6	4	8	34	67	6.08	√
30	2	7	12	1	2	42	92	11.89	√
30	2	2	15	0	0	43	68	7.20	√
30	5	6	15	1	2	45	92	12.33	√
30	5	1	13	0	0	42	66	6.98	√
30	10	49	2	5	33	59	564	127.65	$\Delta > 10^6$
40	2	1	18	0	0	54	90	15.70	√
40	2	4	25	1	2	56	132	26.27	√
40	10	58	12	4	13	63	662	268.39	TL > 250
40	10	2	9	0	0	50	65	10.97	√
40	15	3	7	2	5	50	98	15.48	√
50	2	8	28	4	8	70	140	52.24	√
50	2	40	30	3	6	75	368	253.50	TL > 250
50	10	7	22	0	0	71	113	36.43	√
50	10	55	11	3	13	72	377	253.37	TL > 250
50	20	2	8	3	6	61	88	20.68	√
60	2	30	30	2	4	90	327	249.00	$\Delta > 10^6$
60	2	14	32	2	4	91	271	171.69	A > 15000
60	15	28	16	4	28	90	404	247.99	A > 15000
60	15	12	15	4	22	87	296	181.39	A > 15000
60	25	23	6	5	20	86	388	250.83	TL > 250

√ : problème résolu.

TABLEAU 5.2

Problèmes euclidiens

n = nombre de villes	p = nombre de dépôts	nombre de coupes de Gomory engendrées	nombre de sous-tours de type 1 rencontrés	nombre de sous-tours de type 2 rencontrés	nombre de contraintes de (5.5) engendrées	taille maximum de l'inverse	nombre d'itérations du simplex	temps de résolution	diagnostic
20	2	11	18	0	0	36	87	7.17	√
20	2	5	12	0	0	35	55	3.22	√
20	3	2	9	1	2	30	41	2.23	√
20	3	8	10	4	8	38	77	6.56	√
20	5	61	6	7	20	38	454	63.08	Δ > 10^6
30	2	5	25	0	0	50	68	9.14	√
30	2	5	22	0	0	48	63	8.78	√
30	5	49	14	2	6	53	461	136.38	Δ > 10^6
30	5	58	13	0	0	42	66	6.98	√
30	10	63	5	15	47	66	719	190.85	Δ > 10^6
40	2	4	37	0	0	63	90	19.16	√
40	2	13	29	0	0	65	154	41.17	√
40	10	6	12	2	4	54	76	15.62	√
40	10	15	14	6	20	66	170	51.30	√
40	15	3	10	1	2	49	82	13.47	√
50	2	39	54	0	0	88	379	250.11	TL > 250
50	2	10	33	0	0	76	144	53.47	√
50	10	33	25	7	21	79	334	191.59	Δ > 10^6
50	10	41	27	5	10	77	377	255.36	TL > 250
50	20	2	8	3	6	61	88	20.73	√
60	2	17	50	2	4	97	261	153.48	√
60	2	11	49	0	0	94	156	72.77	√
60	15	19	18	2	7	88	215	138.50	A > 15000
60	15	23	20	5	11	90	214	148.64	A > 15000
60	25	1	6	0	0	62	82	24.39	√

√ : problème résolu.

CHAPITRE 6

LE PROBLEME DES m VOYAGEURS DE COMMERCE AVEC DEPOT VARIABLE (v-m-PVC)

6. LE PROBLEME DES m VOYAGEURS DE COMMERCE AVEC DEPOT VARIABLE (v.m-PVC)

6.1 Introduction

Reprenons le contexte du problème des m voyageurs de commerce (m-PVC). Comme on l'a déjà mentionné au chapitre 3, la localisation précise du dépôt parmi les n villes du m-PVC influence le coût de la solution optimale lorsque le nombre de commis utilisés est supérieur à 1. Ce chapitre concerne la construction d'une méthode exacte trouvant simultanément la localisation optimale du dépôt ainsi que les m tours de coût total minimal qui lui sont associés. Nous dénommerons ce problème par le problème des m voyageurs de commerce avec dépôt variable et il sera abrégé par v.m-PVC. Nous en donnerons une définition plus précise en 6.2.

Le problème étudié ici constitue un cas particulier des problèmes de localisation et d'expédition (PLE) dont les caractéristiques générales ont été données en 1.1. Depuis quelques années les problèmes de type PLE étudiés dans la littérature concernent principalement la localisation optimale d'un ou plusieurs dépôts de façon à minimiser:

1. la somme des distances radiales[1] du (des) dépôt(s) aux villes (clients) [66,99,106,110];
2. la distance radiale maximale du (des) dépôt(s) aux clients [30,56,112].

Le lecteur est référé à [39,81,82] pour des bibliographies exhaustives sur ces types de problèmes.

Le premier des deux modèles précédents n'est vraiment utile que si chaque commis ne visite qu'une seule ville dans sa route. Le second sert surtout à la localisation de services d'urgence. Aucun

---

(1) Longueur de l'aller-retour du dépôt à la ville habituellement pondérée en fonction de la demande de celle-ci ou de la fréquence à laquelle on doit la servir.

de ces deux modèles ne tient compte du fait que, dans plusieurs situations pratiques, un commis peut et a avantage d'un point de vue économique à visiter plus d'une ville avant de revenir au dépôt. Les problèmes de "routing"[1] eux, à l'inverse des problèmes de localisation, permettent ce type de solution mais présupposent que le(s) dépôt(s) est (sont) déjà localisé(s). La relation très étroite entre ces deux types de problèmes a été signalée par plusieurs auteurs, entre autres:

> Sussams [116]: "The best set of vehicle routes to serve a given area is not necessarily that which minimizes the distance travelled. Efficient routing is inextricably bound up with depot siting."
>
> Rand [101] : "Many practitioners are aware of the danger of sub-optimizing by separating depot location from vehicle routing."

De telles remarques, rencontrées à plusieurs endroits dans la littérature, nous ont incités à entreprendre l'étude exposée ici, d'autant plus que très peu d'articles concernent des problèmes où localisation et "routing" sont combinés et qu'à notre connaissance toutes les méthodes présentées sont heuristiques. Ceci est probablement dû à l'opinion généralement admise qu'ils sont trop difficiles à résoudre optimalement pour l'état de la science. Rand [101] a émis le commentaire suivant: "Unfortunately, it will rarely be practicable to determine depot locations using vehicle scheduling packages because of the tremendous increase in computer time."

Madsen [86] a récemment publié un article contenant un résumé des travaux parus dans la littérature concernant les problèmes de type PLE. Il constate lui aussi la rareté de méthodes exactes pour les PLE. Nous nous contenterons ici de résumer très brièvement la littérature concernant les problèmes de type PLE.

Un article écrit par Christofides et Eilon [19] a fortement

---

(1) PVC, m-PVC, c.m-PVC, etc.

influencé la recherche sur les problèmes de type PLE. Dans cet article, les auteurs présentent la formule (6.1) ci-après reliant la longueur[1] totale des tours, notés D_0 dans (6.1), en fonction:

1. du nombre moyen de villes par tour, dénoté par F dans (6.1);
2. de la taille de l'aire servie par le dépôt, notée a dans (6.1);
3. de la somme des distances radiales, notée D_r dans (6.1);

$$D_0 = \frac{A/D_r}{F} + B\sqrt{a}\sqrt{D_r} \qquad (6.1)$$

où A et B sont des constantes à déterminer.

Un heuristique construit par Christofides et Eilon [18] sert au calcul de D_0 pour plusieurs problèmes-tests différents. Ceux-ci sont de la forme d'un c.m-PVC où la restriction de capacité est équivalente à imposer un nombre maximum de villes par tour[2]. Par la suite les constantes A et B sont déterminées par analyse de corrélation multiple. Trois positions différentes du dépôt ont été testées dans un carré (100 x 100) et le coefficient de corrélation obtenu avec (6.1) a toujours été supérieur à 0.99. Christofides et Eilon soulignent à propos de (6.1) que cette formule semble indiquer qu'il existe une relation très bien définie entre D_0, D_r et F et que cette relation semble peu influencée par la position du dépôt. Le choix du meilleur dépôt à l'aide de (6.1) conduit à un problème de type Weber [125].

Après la parution de l'article de Christofides et Eilon, plusieurs autres ont été présentés où la partie localisation des problèmes de type PLE est premièrement résolue par l'utilisation de formules similaires à (6.1) et la partie "routing" est résolue à l'aide d'heuristiques pour le PVC. Lorsqu'il y a plus d'un dépôt, des techniques ont été développées, entre autres par Cooper [26] et Rapp [104], de façon à déterminer simultanément la localisation des dépôts ainsi que les clients qui

---

(1) Nous avons surtout utilisé le terme coût au lieu de longueur ou distance dans ce texte bien que au niveau des tests sur ordinateur, ces coûts (c_{ij}) sont en fait des distances entre 2 points dans le plan.

(2) Selon la notation du chapitre 4, un nombre maximum de villes par tour est imposée avec $d_i = 1$ ($i = 2,\ldots,n$) et $D = F$.

leur sont associés[1]. Le coût de servir un client par un dépôt donné est supposé connu et est approximé habituellement, soit par distances radiales, soit à l'aide de relations telles (6.1). On pondère quelquefois un tel coût par la quantité de service exigée par le client. Une fois les dépôts localisés et les clients associés déterminés, les problèmes de "routing" résultants sont résolus par un heuristique. Soulignons que dans notre chapitre 5 nous résolvons un problème similaire de façon exacte.

L'article de Webb [124] va à l'encontre des résultats de Christofides et Eilon [19]. Webb illustre à l'aide d'exemples que l'utilisation de fonctions simples, telle (6.1), pour approximer le choix particulier d'un dépôt peut conduire à des résultats erronés. Comme on le verra, nos résultats obtenus à l'aide d'une méthode exacte démontrent que la localisation du dépôt influence grandement la longueur totale des m tours et sont donc à première vue en contradiction avec ceux de Christofides et Eilon. Notons toutefois que nous n'avons pas imposé de nombre maximum de villes par tour comme c'est le cas dans l'article de Christofides et Eilon.

6.2 Définition et formulation du v.m-PVC

6.2.1 Définition et notation

Encore ici $N = \{1,\ldots,n\}$ est l'ensemble de villes qui doivent être visitées par les m commis. Pour une raison que nous verrons plus loin, on suppose ici que m est fixé a priori et à partir de maintenant m est considéré comme une constante. De plus on suppose que $m > 1$ car, si $m = 1$, le v.m-PVC se réduit alors au PVC. $C = (c_{ij})$ dénote la matrice des coûts de passage d'une ville à une autre. On suppose dans ce chapitre que C est symétrique[2]. L'extension de notre formulation du v.m-PVC au cas asymétrique s'effectue simplement et ne sera pas donnée ici.

---

(1) Lorsqu'on permet qu'aucun tour ne soit associé à un dépôt ($m_k \geq 0$ au lieu de $m_k > 1$), le (p,m)-PVC considéré au chapitre 5 est très similaire aux problèmes traités par Rapp [104] et Cooper [26].

(2) Dans la suite du chapitre v.m-PVC signifie v.m-PVC symétrique.

La visite des villes par les m commis est assujettie aux mêmes contraintes que dans un m-PVC, c'est-à-dire:

1. chaque ville autre que le dépôt n'est visitée qu'une seule fois par un seul commis et le dépôt est "visité" m fois;
2. le tour de chacun des commis débute et se termine au même dépôt.

Contrairement au m-PVC cependant, le dépôt n'est pas connu à l'avance mais doit être choisi simultanément à la construction des m tours qui lui seront associés. Pour plus de généralité dans notre définition du v.m-PVC, on suppose l'existence d'un ensemble $K \subseteq N$ dans lequel le dépôt doit être déterminé. Toute ville de K qui ne servira pas de dépôt devient alors comme une autre ville de N-K et doit être visitée une fois.

L'objectif à minimiser dans le v.m-PVC contient deux termes: le coût total des m tours plus le coût "d'opération" du dépôt choisi que nous dénoterons par f_k [1] si la ville $k \in K$ est utilisée comme dépôt.

Lorsque le dépôt est connu à l'avance, le v.m-PVC se réduit au m-PVC avec $m > 1$. En termes de la théorie des graphes on peut définir le v.m-PVC symétrique comme la recherche du recouvrement de poids minimal des noeuds d'un graphe non-orienté par m cycles élémentaires disjoints sauf en un seul sommet non spécifié à l'avance. On dénotera par $G = (N,A)$ le graphe non-orienté associé au v.m-PVC.

6.2.2 Formulation du v.m-PVC

La formulation du v.m-PVC présentée ici est similaire à celle que nous avons mise au point pour le m-PVC. En plus des variables x_{ij} $(i < j)$ définies de la façon habituelle, nous aurons besoin ici d'une variable binaire y_k pour chaque ville dans K. Cette variable prendra la valeur 1 si la ville k sert de dépôt et la valeur 0 sinon. Le modèle suivant est valide pour le v.m-PVC:

---

(1) f_k peut représenter aussi un coût de construction d'un dépôt dans la ville k. Les coûts f_k et c_{ij} doivent couvrir la même période de temps.

$$
(P)\quad
\begin{cases}
\text{Min } Z = \sum\limits_{i<j} c_{ij}\, x_{ij} + \sum\limits_{k\in K} f_k\, y_k & (6.2)\\[2ex]
\underline{\text{assujettie à}}: & \\[2ex]
\sum\limits_{k\in K} y_k = 1 & (6.3)\\[2ex]
\sum\limits_{i<k} x_{ik} + \sum\limits_{j>k} x_{kj} = 2 + 2(m-1)y_k \qquad (k \in K) & (6.4)\\[2ex]
\sum\limits_{i<k} x_{ik} + \sum\limits_{j>k} x_{kj} = 2 \qquad (k \in N-K) & (6.5)\\[2ex]
\sum\limits_{\substack{i\in S,\, j\in S\\ i<j}} x_{ij} \le |S| - 1 + (m-1) \sum\limits_{k\in(S\cap K)} y_k^{(1)} \qquad (S \subset N) & (6.6)\\[2ex]
x_{ij} = 0,1,2 \qquad (i \in K,\ i < j) & (6.7)\\[2ex]
x_{ij} = 0,1 \qquad (i\in N-K,\ i < j) & (6.8)\\[2ex]
y_k = 0,1 \qquad (k \in K) & (6.9)
\end{cases}
$$

La fonction objective de P, équation (6.2), évalue le coût total des m tours plus le coût d'opération du dépôt. On peut diviser les contraintes de P en 4 types distincts:

1. la contrainte (6.3) a pour but d'imposer le choix d'un seul dépôt parmi les villes de K;
2. les contraintes (6.4) et (6.5) assurent que le dépôt choisi aura un degré de 2m tandis que les autres villes auront un degré de 2; on les appelle les contraintes de degré pour le v.m-PVC;
3. les contraintes (6.7) jusqu'à (6.9) sont les contraintes de définitions sur les variables de P;

---

(1) Si $S\cap K = \phi$, on suppose que $\sum\limits_{k\in(S\cap K)} y_k = 0$.

4. les contraintes (6.6) constituent les contraintes d'élimination de sous-tours illégaux pour le v.m-PVC.

Les commentaires suivants s'imposent au sujet de ces contraintes. Puisqu'on ne connaît pas a priori la ville qui servira de dépôt, toute variable x_{ij} où $i \in K$ doit pouvoir prendre la valeur 2 afin de modéliser un aller-retour du dépôt à la ville j si la ville i sert de dépôt. De plus, si m était variable, la contrainte (6.4) et les contraintes (6.6) seraient non-linéaires. Nous devons donc fixer m à l'avance pour que P soit un modèle linéaire.

Comme dans la majorité des modèles mathématiques concernant des problèmes de type PEV, les contraintes de degré ne suffisent pas à empêcher la formation de sous-tours illégaux habituellement caractérisés[1] par le fait que l'ensemble des villes les constituant ne contient aucun dépôt. A l'inverse de tels problèmes toutefois, l'ensemble des solutions du v.m-PVC, satisfaisant aux contraintes de degré mais contenant des sous-tours illégaux, n'est pas prédéterminé à l'avance mais varie en fonction du dépôt choisi, c'est-à-dire en fonction des variables y_k. Cette dépendance doit être incluse dans toute contrainte d'élimination de sous-tours illégaux mise au point pour le v.m-PVC. Les contraintes de (6.6) que nous avons élaborées pour le v.m-PVC illustrent comment on peut tenir compte du fait que le dépôt est variable et leur validité peut être prouvée de la façon suivante. Considérons un ensemble de valeurs pour les variables x_{ij} et y_k satisfaisant à (6.3), (6.4), (6.5) et à (6.7) jusqu'à (6.9). Soit $S \subset N$ tel que $2 \leq |S| \leq n-2$ et $p \in K$ tel que $y_p = 1$. Supposons tout d'abord que $p \notin S$. La contrainte particulière de (6.6) correspondant à S s'écrit alors:

$$\sum_{\substack{i \in S, j \in S \\ i<j}} x_{ij} \leq |S| - 1 \qquad (6.10)$$

Si la solution particulière de P que nous considérons forme un sous-tour dans S, cette solution est irréalisable pour le v.m-PVC et il est clair

---

(1) En l'absence de contraintes de capacités, contraintes de fenêtres, etc., voir chapitre 4.

qu'elle ne satisfait pas à (6.10). En fait (6.10) est de la forme des contraintes d'élimination de sous-tours illégaux mises au point par Dantzig et al. [28] et on a déjà expliqué le modus operandi de telles contraintes au chapitre 2. Si par contre aucun sous-tour n'est formé dans S, (6.10) est alors satisfaite car, dans ce cas, |S|-1 est une borne supérieure sur le membre gauche de (6.10).

Supposons maintenant que $p \in S$. Si les m commis visitaient uniquement des villes de S, on aurait alors:

$$\sum_{\substack{i\in S, j\in S \\ i<j}} x_{ij} = |S| - 1 + m = |S| - 1 + m \sum_{k\in(S\cap K)} y_k \qquad (6.11)$$

Mais puisque $S \neq N$, il est impossible que les m tours soient tous contenus dans S. On peut donc écrire:

$$\sum_{\substack{i\in S, j\in S \\ i<j}} x_{ij} < |S| - 1 + m \qquad (6.12)$$

Par (6.12), on peut conclure que:

$$\sum_{\substack{i\in S, j\in S \\ i<j}} x_{ij} \leq |S| - 1 + (m-1) \qquad (6.13)$$

et puisque $p \in S$:

$$\sum_{\substack{i\in S, j\in S \\ i<j}} x_{ij} \leq |S| - 1 + (m-1) \sum_{k\in(S\cap K)} y_k \qquad (6.14)$$

Les constatations précédentes montrent que si une solution de P contient un sous-tour illégal alors cette solution ne satisfait pas à une des contraintes de (6.6); par contre, si une solution de P contient uniquement des sous-tours légaux, alors (6.6) est satisfaite par cette solution. Les contraintes (6.6) sont donc valides pour le v.m-PVC. Soulignons que pour le m-PVC symétrique les sous-tours illégaux contenant exactement 2 villes sont implicitement éliminés par le choix des varia-

bles du modèle, ce qui n'est pas le cas pour le v.m-PVC. Le dépôt étant inconnu à l'avance, toute variable x_{ij} avec $i \in K$ peut éventuellement prendre la valeur 2 et dans (6.6) on a $2 \leq |S| \leq n-2$.

Les contraintes (6.6) généralisent les contraintes (2.24) de Dantzig et al. pour le PVC au cas du v.m-PVC dans ce sens que si l'ensemble S considéré ne contient pas le dépôt alors (6.6) est identique à (2.24). On a vu au chapitre 2 que (2.24) possède une forme équivalente, (2.26), où les variables considérées "vont" de S à $\bar{S}$[1] au lieu de S dans S. Dans le cas du v.m-PVC aussi, il existe une forme équivalente de (6.6) où les variables "vont" de S dans $\bar{S}$ au lieu de S dans S. Ce sont les contraintes (6.15) suivantes généralisant (2.26) au cas du v.m-PVC:

$$\sum_{\substack{i\in S, j\in \bar{S} \\ \text{ou} \\ j\in S, i\in \bar{S} \\ i<j}} x_{ij} \geq 2\Big(1 - \sum_{k\in(S\cap K)} y_k\Big) \qquad (S \subset N) \quad (6.15)$$

La validité de (6.15) pour le v.m-PVC est simple à prouver[2]:

1. si $\sum_{k\in(S\cap K)} y_k = 0$, alors (6.15) est équivalent à (2.26), ce qui est valide pour éliminer un sous-tour ne contenant pas le dépôt;

2. si $\sum_{k\in(K\cap S)} y_k = 1$ alors le membre droit de (6.15) est nul et constitue une borne inférieure valide sur une somme de variables non-négatives.

L'équivalence entre (6.15) et (6.6) se prouve de façon similaire: si le dépôt ne fait pas partie de $(S \cap K)$ alors l'équivalence découle du théorème 2.2; si au contraire le dépôt est élément de $(S \cap K)$ alors (6.15) et (6.6) sont toujours satisfaites.

Remarquons en dernier lieu que P se réduit à P2 donné en page 87

---

(1) Plus précisément, les arêtes associées aux variables sont incidentes à un sommet de S et à un sommet de $\bar{S}$.

(2) On suppose encore ici que (6.3), (6.4), (6.5) et (6.7) jusqu'à (6.9) sont satisfaites par les x_{ij} et y_k.

pour le m-PVC lorsque le dépôt est fixé à l'avance. En effet, supposons que $y_1 = 1$ et que $K = \{1\}$. On a alors que:

1. la contrainte (6.3) peut être éliminée;
2. (6.4) devient:

$$\sum_{j=2}^{n} x_{1j} = 2m$$

3. (6.5) devient:

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2 \qquad (k = 2,\ldots,n)$$

4. (6.6) devient:

$$\sum_{\substack{i\in S, j\in S \\ i<j}} x_{ij} \leq |S| - 1 \qquad (S \subseteq N-\{1\})$$

5. (6.7) devient:

$$x_{1j} = 0,1,2 \qquad (j = 2,\ldots,n)$$

6. (6.8) devient:

$$x_{ij} = 0,1 \qquad (2 \leq i<j \leq n)$$

7. (6.9) peut être éliminée.

6.3 Une méthode exacte de résolution du v.m-PVC

Puisque le v.m-PVC se réduit au m-PVC lorsque le dépôt est connu, une méthode triviale de résolution du v.m-PVC, telle que suggérée par Lawrence et Pengilly [80], consiste à résoudre n fois un m-PVC et de choisir ensuite le dépôt conduisant aux m tours de coût total le plus petit. Comme on le verra par la suite, une telle façon de procéder s'avère de beaucoup moins efficace que l'approche directe de résolution du v.m-PVC que nous avons mise au point.

Notre algorithme pour le v.m-PVC est basé sur le modèle P donné

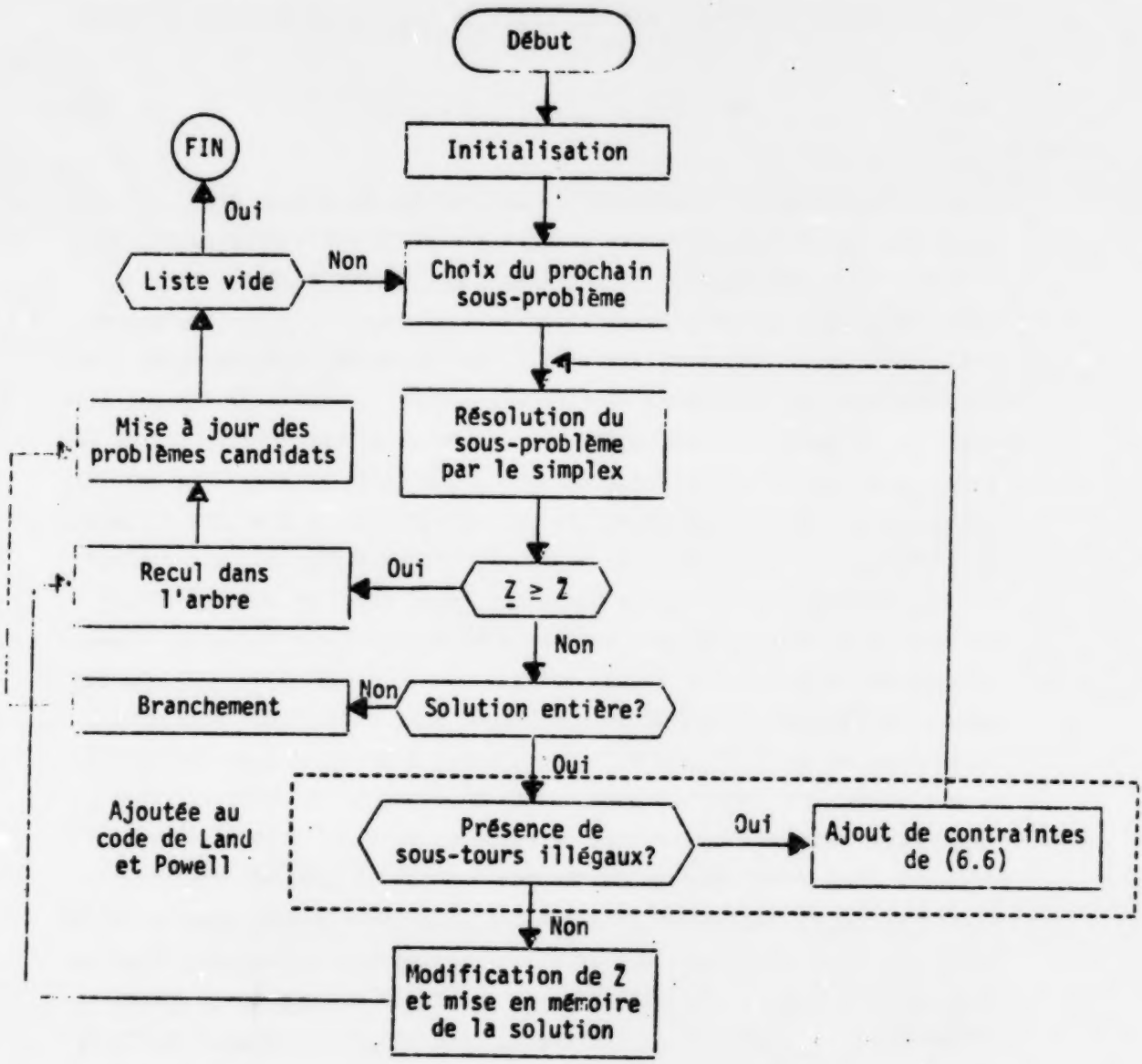
en 6.2.2 pour ce problème. Ce modèle ne peut être résolu directement même pour de petites valeurs de n car le nombre de contraintes dans (6.6), donné par (6.16) et dénoté par NB, croît de façon exponentielle en terme de n:

$$NB = \binom{n}{2} + \ldots + \binom{n}{n-2} = O\left(2^n\right) \qquad (6.16)$$

Une certaine forme de relaxation des contraintes doit donc être utilisée. L'approche que nous avons choisie pour le v.m-PVC est similaire à celles utilisées pour résoudre les autres extensions du PVC étudiées dans ce texte: les contraintes d'élimination de sous-tours illégaux, notamment (6.6), sont initialement relaxées puis, par la suite, quelques-unes d'entre elles sont explicitement réintroduites dans le modèle de façon à éliminer les solutions irréalisables rencontrées. Il est inutile ici de répéter les avantages et les désavantages d'une telle méthode. Le chapitre 3 contient une discussion détaillée sur ce sujet et le chapitre 4 complète la discussion lorsqu'un algorithme de "Branch-and-Bound" est utilisé. De plus, la prise de certaines décisions types que l'on doit effectuer pour implanter sur ordinateur une telle méthode est facilitée par l'expérience acquise au cours de nos travaux sur le m-PVC et le c.m-PVC, les idées sous-jacentes à ces décisions seront donc explicitées brièvement. L'organigramme de la figure 6.1 décrit notre algorithme pour le v.m-PVC. On peut poser les commentaires suivants à propos de ce dernier. Tout d'abord une méthode d'hyperplans de coupe pourrait en théorie être utilisée, au lieu d'une méthode de "Branch-and-Bound" pour maintenir l'intégrité et ce de façon similaire à notre algorithme NORMAL pour le m-PVC. Toutefois, pour la plupart des problèmes testés sur ordinateur, la solution optimale n'a pas été atteinte par une telle méthode car l'ajout d'hyperplans de coupe a fait croître le déterminant de la base de façon très rapide. Il semble que la présence des variables y_k dans le modèle, modifiant la structure des coefficients des contraintes, fait en sorte que le problème n'est plus presque unimodulaire comme c'est le cas pour le PVC ou le m-PVC. L'utilisation d'une méthode d'hyperplans de coupe pour le v.m-PVC a donc été rejetée.

FIGURE 6.1

Organigramme de notre algorithme pour le v.m-PVC



Note : $\bar{Z}$ = coût de la meilleure solution à date

$\underline{Z}$ = coût du sous-problème présentement considéré (= si celui-ci est irréalisable).

Les contraintes de degré ainsi que (6.3) sont présentes dans notre relaxation initiale du v.m-PVC car l'expérience des chapitres 3 et 4 a prouvé que de telles contraintes tendent à être presque toutes effectives dans la solution optimale des problèmes de type PEV. Les contraintes (6.6) ont été jugées préférables à (6.15) à cause du fait qu'en moyenne elles engendrent un moins grand nombre de coefficients. La politique d'ajout des contraintes de (6.6) dans le modèle est identique à celle utilisée pour le m-PVC: une contrainte de (6.6) est adjointe au modèle pour chaque sous-tour illégal rencontré. Nous avons décidé d'atteindre l'intégrité avant de tester si la solution contient des sous-tours illégaux. La raison en est que la notion de sous-tour illégal n'est plus clairement définie dans le cas du v.m-PVC lorsque la solution n'est pas entière, des sommets distincts, avec variables y_k fractionnaires, pouvant se retrouver dans des composantes connexes disjointes. La procédure d'étiquetage permettant de déceler la présence de sous-tours illégaux est similaire à celle utilisée dans notre algorithme pour le m-PVC. Il suffit d'ajoindre à cette dernière quelques instructions localisant le dépôt choisi pour ensuite procéder de la façon habituelle. Notons qu'une même sous-routine FORTRAN localise le dépôt, décèle les sous-tours illégaux et ajoute si nécessaire les contraintes de (6.6) appropriées. Il suffit principalement d'adjoindre cette sous-routine au programme BB de Land et Powell [71] pour obtenir notre algorithme pour le v.m-PVC. D'autres modifications ont été aussi effectuées à l'intérieur de ce code pour optimiser l'utilisation de l'espace-mémoire et implanter une nouvelle stratégie de branchement.

Puisqu'une méthode de "Branch-and-Bound" est utilisée pour résoudre le v.m-PVC, nous avons construit une méthode heuristique de résolution de ce problème de façon à avoir en main au début de l'arbre une borne supérieure, notée $\bar{Z}$, de qualité. Si la valeur initiale de $\bar{Z}$ est petite, il est probable que la taille de l'arbre sera réduite. On peut considérer que l'exécution de cet heuristique fait partie de la phase initialisation de l'organigramme de la figure 6.1. Il se décrit de la façon suivante:

Etape 1:

Pour i ∈ K:

1. considérer i comme dépôt;
2. transformer le m-PVC obtenu en un PVC par la transformation d'Eilon et al. [35];
3. résoudre le PVC obtenu par un heuristique 2-opt [84].

Etape 2:

$\bar{Z}$ est choisi comme la plus petite des n bornes obtenues. ■

Notons qu'on aurait pu utiliser un algorithme 3-opt [84]; on y gagne dans la qualité de la borne mais on y perd au point de vue de la rapidité de l'heuristique. De plus la qualité de la borne supérieure initiale dans une méthode de "Branch-and-Bound" joue un rôle moins important que la qualité de la relaxation choisie (voir [40]).

Nous avons expérimenté deux stratégies de branchement différentes pour le v.m-PVC. La première, implantée dans le programme BB, a été décrite en 1.6.1. Avec la seconde stratégie de branchement, dès que des variables y_k fractionnaires sont rencontrées, la variable y_k qui produirait le plus petit accroissement dans l'objectif si elle était fixée à la valeur 1 est fixée à cette valeur dans le prochain problème à considérer. Dans le cas d'une solution où aucune des variables y_k n'est fractionnaire, le premier critère de branchement s'applique. L'avantage d'un tel branchement provient du fait qu'à partir du moment où une des variables y_k est fixée à 1, le problème à résoudre devient un m-PVC, problème beaucoup plus facile à résoudre et si la variable y_k fixée à 1 a été bien choisie[1], il est probable qu'une borne supérieure efficace sera obtenue rapidement. Les figures (6.2) et (6.3) illustrent ces deux stratégies sur un v.m-PVC euclidien où n = 20 et m = 3. On note que pour ce problème la deuxième stratégie de branchement s'est avérée supérieure à la première, la variable associée au dépôt optimal, notamment y_{10}, ayant été fixée à la valeur 1 au tout début de l'arbre de l'énumération.

---

(1) Le meilleur choix est évidemment la variable correspondant au dépôt optimal.

FIGURE 6.2

Première stratégie de branchement

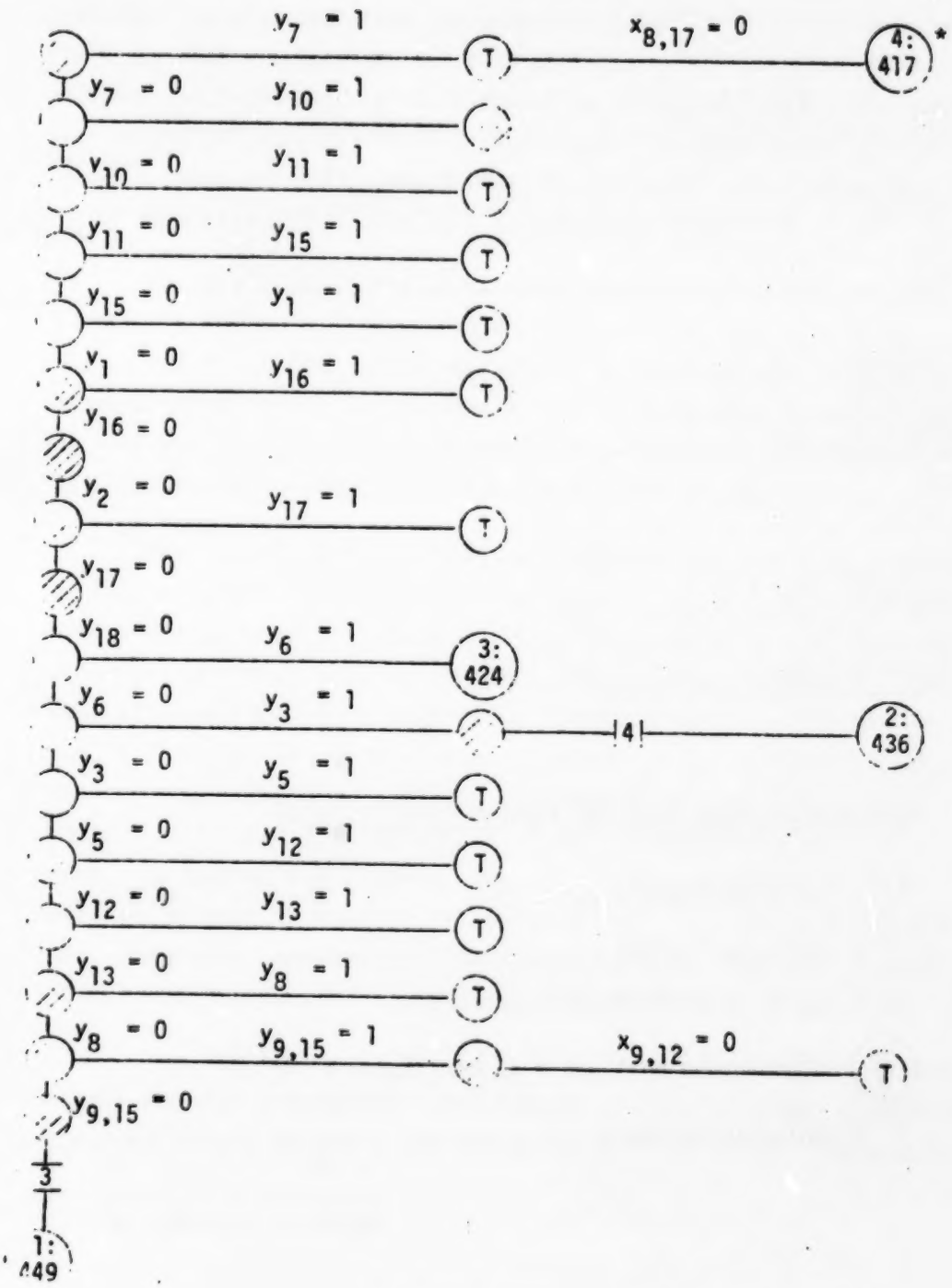
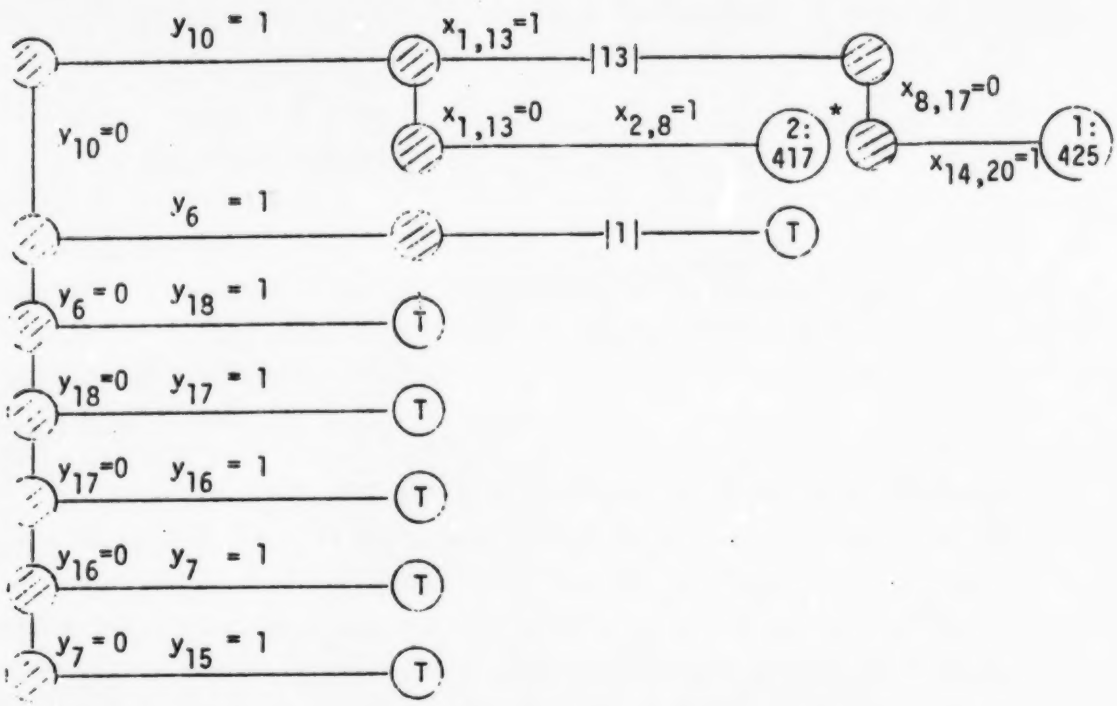


FIGURE 6.3

Deuxième stratégie de branchement



Notation utilisée dans les figures (6.2) et (6.3)

- solution fractionnaire
- (T) queue de l'arbre
- –|p|– ajout de p contraintes de (6.6)
- (i: z) i^e colution réalisable de coût total Z
- ( )* solution optimale

6.4 Résultats sur ordinateur

Les problèmes-tests sont engendrés aléatoirement selon la procédure décrite en 1.2. Il s'agit de v.m-PVC symétriques où la valeur de m a été choisie de façon aléatoire entre 2 et 5 à l'aide d'une distribution uniforme. De plus dans tous les problèmes testés sur ordinateur, l'ensemble K des dépôts potentiels est posé égal à N, l'ensemble des villes du problème. A notre avis, plus le nombre de dépôts potentiels est grand plus le v.m-PVC sera difficile à résoudre. Nous avons donc considéré pour nos tests le cas le plus difficile. Puisque G est complet le nombre de variables dans le modèle à résoudre est $n+(\frac{1}{2}n(n-1))$.

Le tableau 6.1 ci-après illustre une remarque mentionnée précédemment à l'effet qu'il est préférable d'utiliser notre algorithme pour le v.m-PVC plutôt que de résoudre n fois un m-PVC par notre algorithme RENVERSE décrit au chapitre 3. Un problème euclidien de 20 villes avec m = 3 et un problème non-euclidien avec m = 5 ont été considérés. La première stratégie de branchement a été utilisée.

Pour le problème euclidien, notre algorithme a pris 4.71 secondes pour atteindre la solution optimale tandis que la résolution de 20 3-PVC a nécessité 25.37 secondes. En ce qui concerne le problème non-euclidien, les temps obtenus sont 10.5 secondes pour notre algorithme comparativement à 21.96 secondes pour la résolution de 20 5-PVC. De plus, à l'analyse de nos résultats au chapitre 3 sur le m-PVC, on peut facilement se convaincre que notre approche directe au v.m-PVC sera d'autant supérieure à la résolution[1] de n fois un m-PVC que n sera grand. Le tableau 6.1 illustre aussi la grande variabilité du coût total des m tours en fonction du dépôt utilisé. Ce qui constitue d'ailleurs une preuve supplémentaire de l'importance de localiser le dépôt optimal simultanément avec la construction des m tours optimaux.

Dans les tableaux 6.2 jusqu'à 6.5, les deux stratégies de branchement décrites précédemment sont testées sur des problèmes euclidiens et non-euclidiens de taille variant entre 20 et 50 villes. Dans tous

---

(1) Par RENVERSE ou NORMAL.

TABLEAU 6.1

Résolution d'un v.m-PVC et n résolutions d'un m-PVC

Ville utilisée comme dépôt	Problèmes euclidiens (m = 3)		Problèmes non-euclidiens (m = 5)	
	Temps de résolution	Coût optimal	Temps de résolution	Coût optimal
1	.94	342	1.04	339
2	1.22	367	.78	253
3	1.06	336	1.16	401
4	.94	380	.91	371
5	1.23	358	.95	320
6	1.40	427	1.20	275
7	1.31	400	1.17	367
8	1.00	359	1.02	390
9	1.28	351	.94	460
10	1.20	360	1.80	403
11	.98	396	.87	241
12	1.96	364	.80	330
13	1.37	345	1.23	234*(opt)
14	1.24	350	.92	311
15	1.50	359	1.55	565
16	1.15	353	.83	253
17	1.56	347	.92	315
18	1.10	332*(opt)	.88	346
19	1.53	354	1.16	383
20	1.40	357	1.83	351
	Total 25.37		Total 21.96	
Notre algorithme pour le v.m-PVC	4.71	332	10.5	234

ces problèmes les coûts d'ouverture des dépôts sont nuls. L'importance de la statistique sur la taille maximum de l'inverse a été expliquée au chapitre 3. Les conclusions suivantes se dégagent des tableaux 6.2 jusqu'à 6.5:

1. tel que prévu, le v.m-PVC est un problème beaucoup plus difficile à résoudre que le m-PVC; pour des problèmes de taille identique, les temps de résolution sont plus élevés et ont une plus grande variance;

2. la relaxation utilisée s'est avérée toutefois très efficace car relativement[1] peu de contraintes de (6.6) sont engendrées;

3. contrairement au m-PVC, en comparant entre elles les 3^e et 4^e colonnes des tableaux 6.2 jusqu'à 6.5 on s'aperçoit qu'un bon nombre de contraintes de (6.6) engendrées sont ineffectives dans la solution optimale, surtout lorsque n est grand. A toute contrainte de (6.6) est associé un sous-ensemble S de N. Si, lorsque la solution optimale est atteinte, cet ensemble contient le dépôt optimal, il est très probable que la contrainte associée soit ineffective car pour qu'il y ait égalité entre le membre gauche et le membre droit de cette contrainte, il faudrait que les tours des commis soient presque tous inclus dans S;

4. la seconde stratégie de branchement semble un peu supérieure à la première surtout lorsque n est grand;

5. le dépôt heuristique est rarement identique au dépôt optimal et le coût heuristique est souvent loin du coût optimal;

6. le facteur temps est plus limitatif que le facteur mémoire pour cet algorithme[2]. Ceci s'explique par le fait qu'une méthode de "Branch-and-Bound" est utilisée au lieu d'une méthode d'hyperplans de coupe. Ces derniers prennent

---

(1) Par rapport à NB donné en (6.16).

(2) Phénomène inverse dans le cas de notre algorithme pour le m-PVC.

TABLEAU 6.2

Problèmes euclidiens, première stratégie de branchement

n	No. du problème	m	Taille maximum de l'inverse	Nombre total de contraintes engendrées	Nombre de contraintes de (5.6) engendrées	Nombre d'itérations du simplex	Temps de résolution	Coût de la solution optimale (dépôt optimal)	Coût de la solution heuristique (dépôt heuristique)
20	1	3	30	34	13	127	4.71	332 (18)	345 (3)
	2	3	35	47	26	545	27.96	410 (8)	443 (13)
	3	3	32	38	17	359	14.06	417 (10)	474 (9)
	4	3	33	44	23	447	17.78	393 (19)	446 (4)
	5	5	32	37	16	666	22.33	438 (10)	485 (5)
25	1	4	36	50	24	464	25.04	471 (22)	540 (4)
	2	3	37	47	21	172	9.98	440 (14)	476 (15)
	3	3	42	46	20	358	19.12	407 (13)	459 (21)
	4	3	40	49	23	313	19.23	395 (14)	420 (25)
	5	4	41	53	27	1113	70.96	483 (8)	539 (4)
30	1	2	49	72	41	429	48.59	435 (30)	451 (9)
	2	4	46	50	19	525	35.71	445 (4)	503 (17)
	3	4	50	74	43	1422	138.26	487 (3)	522 (24)
	4	4	46	62	31	797	61.46	472 (10)	501 (7)
	5	4	44	52	21	1459	137.15	487 (17)	487 (17)
35	1	4	56	78	42	2625	386.75	541 (8)	581 (29)
	2	5	52	61	25	1083	104.00	486 (27)	548 (2)
	3	4	53	63	28	894	84.51	482 (7)	560 (32)
	4	3	52	55	19	412	41.26	486 (8)	570 (28)
	5	3	58	69	33	946	119.82	468 (5)	524 (29)
40	1	3	62	82	41	736	140.40	469 (6)	537 (5)
	2	3	68	81	40	3392	694.62	542 (4)	593 (23)
50	1	4	80	101	50	2596	935.83	565 (10)	616 (49)
	2	3	81	112	61	2128	688.34	505 (26)	556 (15)

TABLEAU 6.3

Problèmes euclidiens, deuxième stratégie de branchement

n	No. du problème	m	Taille maximum de l'inverse	Nombre total de contraintes engendrées	Nombre de contraintes de(5.6) engendrées	Nombre d'itérations du simplex	Temps de résolution	Coût de la solution optimale (dépôt optimal)	Coût de la solution heuristique (dépôt heuristique)
20	1	3	30	33	12	278	9.35	332 (18)	345 (3)
	2	3	34	44	23	422	20.21	410 (8)	443 (13)
	3	3	30	35	14	258	9.13	417 (10)	474 (9)
	4	3	29	36	15	241	9.79	292 (19)	446 (4)
	5	5	32	36	15	907	30.61	438 (10)	485 (5)
25	1	4	37	40	14	264	11.34	471 (22)	540 (14)
	2	3	38	44	18	237	12.18	440 (14)	476 (15)
	3	3	41	45	19	209	10.51	407 (15)	450 (21)
	4	3	39	45	19	220	11.64	395 (14)	420 (25)
	5	4	43	49	23	792	43.69	483 (8)	539 (4)
30	1	2	49	73	42	365	37.84	435 (9)	451 (9)
	2	4	45	50	19	1166	78.89	445 (4)	503 (17)
	3	4	51	65	34	1572	143.07	487 (3)	522 (24)
	4	4	54	68	37	11180	979.85	472 (10)	501 (7)
	5	4	45	52	21	1224	83.86	487 (17)	487 (17)
35	1	4	56	76	40	11394	1097.44	541 (8)	581 (29)
	2	5	55	62	26	2009	217.11	486 (27)	548 (2)
	3	4	55	60	24	1181	110.34	482 (7)	560 (32)
	4	3	55	66	30	598	56.43	486 (8)	570 (28)
	5	3	55	65	29	478	55.84	468 (5)	524 (29)
40	1	3	63	86	45	1687	355.34	469 (6)	537 (5)
	2	3	*	*	*	9736	*	565 (34)	593 (23)
50	1	4	80	99	48	2280	616.50	565 (10)	616 (49)
	2	3	88	102	51	2991	726.78	505 (44)	556 (15)

(*) Temps limite atteint (550 sec.)

TABLEAU 6.4

Problèmes non-euclidiens, première stratégie de branchement

n	No. du problème	m	Taille maximum de l'inverse	Nombre total de contraintes engendrées	Nombre de contraintes de (5.6) engendrées	Nombre d'itérations du simplex	Temps de résolution	Coût de la solution optimale (dépôt optimal)	Coût de la solution heuristique (dépôt heuristique)
20	1	5	27	37	16	345	10.50	234 (13)	254 (13)
	2	2	26	28	7	62	1.79	207 (14)	291 (5)
	3	5	28	39	18	398	12.83	260 (7)	359 (15)
	4	2	28	34	13	46	1.31	185 (5)	313 (5)
	5	3	30	34	13	240	10.50	179 (2)	276 (18)
25	1	2	32	37	11	49	1.97	213 (10)	300 (25)
	2	5	36	45	19	952	45.06	203 (16)	277 (24)
	3	3	33	40	14	99	4.39	180 (12)	258 (23)
	4	3	37	46	20	338	20.72	193 (2)	247 (24)
	5	3	35	42	16	492	26.62	219 (20)	253 (20)
30	1	2	43	49	18	132	8.15	197 (26)	239 (20)
	2	3	43	50	19	547	39.50	195 (8)	286 (18)
	3	4	41	47	16	740	52.34	235 (17)	280 (17)
	4	4	42	53	22	636	46.88	253 (19)	293 (19)
	5	5	47	67	36	1640	125.40	253 (8)	341 (4)
35	1	4	53	80	44	2270	263.48	192 (17)	289 (32)
	2	3	51	60	24	536	51.25	173 (7)	243 (15)
	3	4	47	55	19	357	32.33	183 (7)	307 (7)
	4	4	52	73	37	2806	333.76	252 (29)	392 (2)
	5	4	50	60	24	1020	114.58	211 (5)	292 (5)
40	1	3	56	74	33	932	109.87	243 (13)	365 (24)
	2	4	59	73	32	1496	197.89	219 (22)	329 (2)
50	1	1	*	*	*	*	*		
	2	2	*	*	*	*	*		

(*) Temps limite atteint (750 sec.)

TABLEAU 6.5

Problèmes non-euclidiens, deuxième stratégie de branchement

n	No. du problème	m	Taille maximum de l'inverse	Nombre total de contraintes engendrées	Nombre de contraintes de (5.6) engendrées	Nombre d'itérations du simplex	Temps de résolution	Coût de la solution optimale (dépôt optimal)	Coût de la solution heuristique (dépôt heuristique)
20	1	5	28	39	18	589	19.27	234 (13)	254 (13)
	2	2	25	28	7	54	1.56	207 (19)	291 (5)
	3	5	26	28	7	639	19.21	260 (7)	359 (15)
	4	2	28	34	13	46	1.33	185 (5)	313 (5)
	5	3	30	34	13	268	10.53	179 (2)	276 (18)
25	1	2	32	37	11	49	1.98	213 (10)	300 (25)
	2	5	35	37	11	1109	51.66	203 (16)	277 (24)
	3	3	33	40	14	70	3.34	180 (12)	258 (23)
	4	3	38	51	25	445	25.28	193 (2)	247 (24)
	5	3	34	39	13	130	6.14	219 (20)	253 (20)
30	1	2	45	50	19	122	8.63	197 (26)	239 (20)
	2	3	43	52	21	215	14.06	195 (8)	286 (18)
	3	4	43	47	16	841	54.56	235 (17)	280 (17)
	4	4	45	56	25	844	61.90	253 (19)	293 (19)
	5	5	42	46	15	1126	75.19	253 (8)	341 (4)
35	1	4	52	72	36	1146	125.52	192 (17)	289 (32)
	2	3	51	60	24	307	28.08	173 (7)	243 (15)
	3	4	53	70	34	1505	160.49	183 (7)	307 (7)
	4	4	57	69	33	1926	182.83	253 (13)	392 (2)
	5	4	51	60	24	1211	117.67	211 (5)	292 (2)
40	1	3	55	66	25	616	69.24	243 (18)	365 (24)
	2	4	56	70	29	559	75.07	219 (22)	329 (2)
50	1	4	74	89	48	1650	321.49	203 (35)	355 (38)
	2	4	74	88	47	2899	546.12	237 (49)	287 (34)

beaucoup d'espace-mémoire et causent souvent des problèmes de stabilité numérique;

7. pour une valeur donnée de n, le v.m-PVC semble d'autant plus difficile à résoudre que m est grand. Le phénomène inverse a été constaté pour le m-PVC. Il semble que pour le v.m-PVC, lorsque m est grand, des sous-tours se forment autour de plusieurs dépôts différents et un plus grand nombre de contraintes de (6.6) doivent être engendrées.

Dans le tableau 6.6, l'impact d'imposer des coûts d'opération non nuls sur les temps de résolution de notre algorithme a été testé avec les deux stratégies de branchement sur dix problèmes de 20 villes dont 5 sont euclidiens et 5 sont non-euclidiens. Les valeurs de f_k engendrées sont du même ordre de grandeur que les c_{ij} de façon à éviter que seuls les f_k servent à déterminer le dépôt optimal.

TABLEAU 6.6

Temps de résolution avec coûts f_k non-nuls sur 10 problèmes de 20 villes

No. du problème	Type	m	Première stratégie de branchement		Deuxième stratégie de branchement	
			Temps	Nombre d'itérations	Temps	Nombre d'itérations
1	Euclidien	3	3.42	100	2.05	63
2		3	18.05	415	8.05	163
3		3	1.05	36	1.05	36
4		3	4.25	104	5.46	139
5		5	17.17	472	17.43	523
6	Non-euclidien	5	14.09	422	14.76	463
7		2	2.24	55	2.37	65
8		5	18.01	561	24.69	770
9		2	1.36	41	1.36	41
10		3	13.88	361	2.50	82

Comme on peut le constater, les temps fournis dans le tableau 6.6 sont similaires à ceux donnés dans les tableaux précédents et la présence de f_k non nuls n'affectent pas la performance de notre méthode. A notre avis, la présence de f_k relativement élevés (petits) par rapport aux c_{ij} ne peut que rendre le v.m-PVC encore plus facile car dans l'énumération les branches de l'arbre avec $y_k = 1$ ($y_k = 0$) pourront être rapidement éliminées pour une ville k avec un coût d'ouverture élevé (petit). Si par contre, les dépôts ont tous un coût d'ouverture du même ordre de grandeur, la localisation du dépôt optimal sera plus difficile car seule la construction des m tours optimaux qui leurs sont associés permettra de les départager entre eux. C'est le cas que nous avons considéré dans tous les problèmes-tests des tableaux 6.2 jusqu'à 6.5.

6.5 Conclusion

Dans ce chapitre nous avons mis au point une nouvelle formulation pour un problème dont l'importance pratique a été soulevée par plusieurs auteurs. A partir de cette formulation nous avons construit ce qui constitue à notre connaissance la première méthode exacte capable de résoudre un problème de type PLE. Les tailles atteintes sont très respectables malgré la grande difficulté de résolution du v.m-PVC[1]. De plus les nouvelles contraintes d'élimination de sous-tours illégaux (6.6) que nous avons élaborées pour le v.m-PVC, généralisant des contraintes du même type bien connues pour le PVC, semblent très "fortes" tel qu'en témoigne le succès de la relaxation que nous avons utilisée. Ceci est important car l'utilisation de telles contraintes dans le cadre d'une autre technique d'optimisation, telle la relaxation lagrangienne, est susceptible de produire aussi de bons résultats.

Nos recherches sur le v.m-PVC, en plus de suggérer de nouvelles approches optimales pour ce problème et de conduire à une meilleure compréhension des caractéristiques de sa solution optimale, ouvrent la voie

(1) Des problèmes de 50 villes dans les cas symétrique euclidien et symétrique non-euclidien ont été résolus.

aussi à de nouvelles recherches sur d'autres extensions du PVC où, par exemple, la localisation de plus d'un dépôt doit être exécutée de façon optimale simultanément avec la construction des tours associés.

CHAPITRE 7

CONCLUSIONS ET EXTENSIONS

7. CONCLUSIONS ET EXTENSIONS

7.1 Conclusion

Nous avons considéré dans cette étude quatre généralisations différentes du problème du voyageur de commerce (PVC). Il s'agit:

1. du problème des m voyageurs de commerce (m-PVC);
2. du problème des m voyageurs de commerce assujettis à une restriction de capacité (c.m-PVC);
3. du problème des m voyageurs de commerce avec plusieurs dépôts ((p,m)-PVC);
4. du problème des m voyageurs de commerce avec dépôt variable (v.m-PVC);

Pour chacun de ces problèmes, nous avons mis au point un modèle mathématique et une méthode exacte de résolution. Nos principaux résultats concernant les modèles établis sont:

1. généralisation des contraintes d'élimination de sous-tours de Dantzig et al. [28] pour tenir compte d'une restriction de capacité ou d'un dépôt variable;
2. construction de contraintes d'élimination de chaînes dans le cas d'un problème avec plusieurs dépôts;
3. application de la décomposition de Benders [9] au modèle de Gavish et Graves [45] du c.m-PVC pour retrouver notre modèle pour ce même problème.

Les contraintes de nos modèles pour les quatre problèmes considérés peuvent être divisées en trois types principaux:

1. contraintes de degré;
2. contraintes d'élimination de sous-tours illégaux;
3. contraintes de définition sur les variables.

Une approche unifiée a été utilisée pour résoudre de tels modèles. il s'agit de relaxer au début de la procédure les contraintes d'élimination de sous-tours illégaux et de résoudre à l'aide de la méthode du simplex le modèle linéaire continu ne contenant que les contraintes de degré. Par la suite, les contraintes relaxées non satisfaites sont réintroduites explicitement dans le modèle et plusieurs phases de réoptimisation s'ensuivent. L'intégrité sur les variables est maintenue soit par des coupes de Gomory soit par une méthode de "Branch-and-Bound".

Une approche de ce type a été utilisée avec succès par plusieurs auteurs dont [27,54,92,98] sur le PVC symétrique. Nos résultats confirment l'efficacité d'une telle procédure pour plusieurs généralisations du PVC de ce type. Ainsi nos algorithmes pour le m-PVC et le c.m-PVC se sont avérés compétitifs sinon supérieurs à ceux déjà connus pour ces mêmes problèmes. En outre des (p,m)-PVC symétriques avec 60 villes et des (v.m-PVC) symétriques contenant 50 villes ont été résolus optimalement. Il s'agit à notre avis d'un résultat important car la littérature ne contient, à notre connaissance, aucun algorithme optimal pouvant résoudre ces deux derniers problèmes. L'opinion généralement admise était qu'ils sont trop difficiles pour être abordés par une méthode exacte.

7.2 Extensions

Un des avantages majeurs du type d'approche que nous avons considérée est sa flexibilité. Son utilisation avec succès sur quatre problèmes différents en fait foi. Il est même possible qu'une telle approche soit efficace pour d'autres généralisations du PVC. En particulier il serait intéressant à notre avis d'envisager la résolution des problèmes suivants dans le cadre d'une telle procédure:

1. localisation de plus d'un dépôt simultanément à la construction de routes pour plusieurs commis;
2. imposition de contraintes de distance maximum sur les routes des commis;

3. le problème dénommé "Subscriber dial-a-ride problem" pour lequel le concept des contraintes d'élimination de chaînes pourrait être utile.

Notons finalement le fait qu'en pratique, la route d'un commis peut quelquefois contenir plus d'une visite à la même ville. Cette modification peut s'appliquer au PVC et à tous les problèmes que nous avons traités. Ainsi, pour le PVC, lorsque le graphe G = (N,A) qui lui est associé n'est pas complet, il pourrait ne pas exister un cycle ou circuit hamiltonien dans G. Un problème intéressant à résoudre, noté ici PCC, serait alors celui de la détermination du cycle ou circuit complet le plus court dans G c'est-à-dire une route passant par tous les sommets de G au moins une fois (et non pas exactement une fois). Remarquons de plus que même si G est complet, la solution optimale du PVC peut conduire à une route plus coûteuse que celle du PCC si la matrice des coûts de transport $C = (c_{ij})$ n'est pas euclidienne.

Nous présentons ici, à titre indicatif et sans entrer dans les détails, le modèle que nous avons développé dans [93] pour le PCC symétrique (non-euclidien).

Soit G' le graphe obtenu à partir de G en éliminant de G toute arête (i,j) qui n'est pas elle-même la chaîne la plus courte entre i et j. On associe comme d'habitude une variable x_{ij} à chaque arête (i,j) de G'. Notre modèle pour le PCC est donc:

(P)

$$\text{Min } Z = \sum_{i<j} c_{ij}\, x_{ij} \qquad (7.1)$$

<u>assujettie à</u>:

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} - 2y_k = 2 \qquad (k = 1,\ldots,n) \qquad (7.2)$$

$$\sum_{\substack{i\in S, j\in \bar{S} \\ j\in S, i\in \bar{S} \\ i<j}} x_{ij} \geq 2 \qquad (S \subset N, 2 \leq |S| \leq n-2) \qquad (7.3)$$

$$\begin{cases} x_{ij} = 0,1,2 & (1 \leq i < j \leq n) \quad (7.4) \\ y_k \geq 0 \text{ et entier} & (k = 1,\ldots,n) \quad (7.5) \end{cases}$$

Nous avons résolu le modèle P à l'aide d'une procédure similaire à celle utilisée tout au long de ce texte. Soit x^*_{ij} la solution optimale de P et G* le graphe obtenu en répétant x^*_{ij} fois chaque arête (i,j) de G'. Puisque chaque noeud de G* possède un degré pair par (7.2) et puisque G* est connexe par (7.3), G* contient au moins un cycle d'Euler et il est facile d'en déterminer un.

Avec notre algorithme, nous avons résolu des problèmes symétriques non-euclidiens contenant 100 villes. Notre modèle et notre algorithme pour le PCC s'adaptent facilement au cas asymétrique. Pour ce type de matrice de coûts de transport, nous avons résolu des problèmes contenant 110 villes.

Permettre plus d'une visite à la même ville dans le contexte du m-PVC, du c.m-PVC, du (p,m)-PVC ou du v.m-PVC engendre des modifications dans nos modèles et dans nos algorithmes qui ne sont pas triviales à déterminer. A notre avis toutefois, la méthode générale que nous avons utilisée s'avérerait tout aussi efficace tel que le suggère les résultats obtenus pour le PCC avec une telle procédure.

REMERCIEMENTS

Je tiens à remercier en tout premier lieu mon directeur de thèse et ami Gilbert Laporte dont la compétence, la disponibilité et les encouragements ont grandement contribué à l'achèvement de cette étude.

Je voudrais aussi rendre hommage à Michèle Tremblay qui a dactylographié ce texte et à Hélène Mercure qui a effectué la programmation des tests sur ordinateur pour le chapitre 5.

Je suis aussi reconnaissant au Gouvernement du Québec et à Transports Canada pour l'aide financière qu'ils m'ont apportée au cours de mes études.

Finalement, je tiens à remercier Michèle, mon épouse qui, par son affection et son amour, a réussi à me motiver tout au long de cette recherche.

BIBLIOGRAPHIE

[1] ALI, A.I., KENNINGTON, J.L.: "The M-Travelling Salesmen Problem: A Duality Based Branch-and-Bound Algorithm", Technical Report OR 80018, Department of Operations Research, Southern Methodist University (1980).

[2] BALAS, E., CHRISTOFIDES, N.: "A Restricted Lagrangean Approach to the Travelling Salesman Problem", Mathematical Programming 21, 19-46 (1981).

[3] BALAS, E., PADBERG, M.W.: "Set Partitioning - A Survey", Combinatorial Optimization, 151-210, Wiley (1979).

[4] BALINSKI, M., QUANDT, R.: "On an Integer Program for a Delivery Problem", Operations Research 12, 300-304 (1964).

[5] BELLMAN, R.: "Dynamic Programming Treatment of the Travelling Salesman Problem", JACM 9, 61-63 (1962).

[6] BELLMORE, M., HONG, S.: "Transformation of Multisalesmen Problem to the Standard Traveling Salesman Problem", JACM 21, 500-504 (1974).

[7] BELLMORE, M., MALONE, J.C.: "Pathology of Traveling Salesman Subtour-Elimination Algorithms", Operations Research 19, 278-307 (1971).

[8] BELLMORE, M., NEMHAUSER, G.: "The Travelling Salesman Problem: A Survey", Operations Research 16, 537-558 (1968).

[9] BENDERS, J.F.: "Partitioning Procedures for Solving Mixed-Variables Programming Problems", Numerische Mathematik 4, 238-252 (1962).

[10] BERGE, C.: "Graphs and Hypergraphs", North-Holland (1976).

[11] BODIN, L., GOLDEN, B.: "Classification in Vehicle Routing and Scheduling", Networks 11, 97-108 (1981).

[12] BODIN, L., GOLDEN, B., ASSAD, A., BALL, M.: "The State of the Art in the Routing and Scheduling of Vehicles and Crews", Report No. UMTA/BMGT/MSS#81-001, College of Business and Management, University of Maryland (1981).

[13] BURKARD, R.E.: "Travelling Salesman and Assignment Problems: A Survey", Annals of Discrete Mathematics 4, 193-215 (1979).

[14] BUXEY, G.M.: "The Vehicle Scheduling Problem and Monte Carlo Simulation", Journal of the Operational Research Society 30, 563-573 (1979).

[15] CARPANETO, G., TOTH, P.: "Some New Branching and Bounding Criteria for the Asymmetric Travelling Salesman Problem", Management Science 26, 736-743 (1980).

[16] CHRISTOFIDES, N.: "The Vehicle Routing Problem", RAIRO (recherche opérationnelle) 10, 55-70 (1976).

[17] CHRISTOFIDES, N.: "The Travelling Salesman Problem", Combinatorial Optimization, 131-149, Wiley (1979).

[18] CHRISTOFIDES, N., EILON, S.: "An Algorithm for the Vehicle Dispatching Problem", Operational Research Quarterly 20, 309-318 (1969).

[19] CHRISTOFIDES, N., EILON, S.: "Expected Distances in Distribution Problems", Operational Research Quarterly 20, 437-443 (1969).

[20] CHRISTOFIDES, N., MINGOZZI, A., TOTH, P.: "The Vehicle Routing Problem", Combinatorial Optimization, 315-338, Wiley (1979).

[21] CHRISTOFIDES, N., MINGOZZI, A., TOTH, P.: "Exact Algorithms for the Vehicle Routing Problem, Based on Spanning Tree and Shortest Path Relaxations", Mathematical Programming 20, 255-282 (1981).

[22] CHRISTOFIDES, N., MINGOZZI, A., TOTH, P.: "Space State Relaxation Procedures for the Computation of Bounds to Routing Problems", Networks 11, 145-164 (1981).

[23] CHRISTOFIDES, N., WHITLOCK, C.: "An LP-Based TSP Algorithm", Imperial College Report OR 78.14 (1978).

[24] CHVATAL, V.: "Edmonds Polytopes and Weakly Hamiltonian Graphs", Mathematical Programming 5, 29-40 (1973).

[25] CLARKE, G., WRIGHT, J.W.: "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points", Operations Research 12, 568-581 (1964)

[26] COOPER, L.: "Heuristic Methods for Location-Allocation Problems", SIAM Review 6, 37-53 (1964).

[27] CROWDER, H., PADBERG, M.W.: "Solving Large-Scale Symmetric Travelling Salesman Problems to Optimality", Management Science 26, 495-509 (1980).

[28] DANTZIG, G.B., FULKERSON, D.R., JOHNSON, S.M.: "Solution of a Large Scale Travelling Salesman Problem", Operations Research 2, 393-410 (1954).

[29] DANTZIG, G.B., RAMSER, J.: "The Truck Dispatching Problem", Management Science 6, 81-91 (1959).

[30] DEARING, P.M., FRANCIS, R.L.: "A One Facility Minimax Location Problem Involving Rectilinear Distances", ORSA Bulletin 20, Sup. 2 B-469 (1972).

[31] DINIC, E.A., KRONRAD, M.A.: "An Algorithm for the Solution of the Assignment Problem (translated from Russian)", Soviet Math. Kokl. 10, 1324-1326 (1969).

[32] DISCENZA, J.H.: "A More Compact Formulation of the Symmetric Multiple Traveling Salesman Problem with Fixed Charges", Networks 11, 73-75 (1981).

[33] EDMONDS, J.: "Maximum Matching and a Polyhedron with 0,1 Vertices", Journal of Research of the National Bureau of Standards Section B 69, 125-130 (1965).

[34] EDMONDS, J.: "Matroids and the Greedy Algorithm", Mathematical Programming 1, 127-136 (1971).

[35] EILON, S., WATSON-GANDY, C.D.T., CHRISTOFIDES, N.: "Distribution Management: Mathematical Modelling and Practical Analysis", Griffin (1971).

[36] FISHER, M., JAIKUMAR, R.: "A Generalized Assignment Heuristic for Vehicle Routing", Networks 11, 109-124 (1981).

[37] FORD, L.R., FULKERSON, D.R.: "Flows in Networks", Princeton University Press (1962).

[38] FOSTER, B.A., RYAN, D.M.: "An Integer Programming Approach to the Vehicle Scheduling Problem", Operational Research Quarterly 27, 367-384 (1976).

[39] FRANCIS, R.L., GOLDSTEIN, J.M.: "Location Theory: A Selective Bibliography", Operations Research 22, 400-410 (1974).

[40] GARFINKEL, R.S.: "Branch and Bound Methods for Integer Programming", Combinatorial Optimization, 1-20, Wiley (1979).

[41] GARVIN, W.M., CRANDALL, H.W., JOHN, J.B., SPELLMAN, R.A.: "Applications of Linear Programming in the Oil Industry", Management Science 3, 407-430 (1957).

[42] GASKELL, T.J.: "Bases for Vehicle Fleet Scheduling", Operational Research Quarterly 18, 281-295 (1967).

[43] GAVISH, B.: "A note on the Formulation of the M-Salesman Traveling Salesman Problem", Management Science 22, 704-705 (1976).

[44] GAVISH, B., Communication Privée, Université de Rochester (1979).

[45] GAVISH, B., GRAVES, S.C.: "The Travelling Salesman Problem and Related Problems", Working Paper No. 7905, Graduate School of Management, University of Rochester (1979).

[46] GAVISH, B., SHLIFER, E.: "An Approach for Solving a Class of Transportation Scheduling Problems", Working Paper No. 7731, Graduate School of Management, University of Rochester (1977).

[47] GAVISH, B., SRIKANTH, K.: "An Optimal Solution Method for the Multiple Travelling Salesman Problem", Working Paper No. 8027, Graduate School of Management, University of Rochester (1980).

[48] GILLET, B.E., JOHNSON, T.: "Multi-Terminal Vehicle-Dispatch Algorithm", Omega 4, 711-718 (1976).

[49] GILLET, B.E., MILLER, L.R.: "A Heuristic Algorithm for the Vehicle-Dispatch Problem", Operations Research 22, 341-349 (1974).

[50] GILMORE, P.C., GOMORY, R.E.: "Sequencing a one State-Variable Machine: a Solvable Case of the Travelling Salesman Problem, Operations Research 12, 655-679 (1964).

[51] GOLDEN, B., BODIN, L., DOYLE, T., STEWART, W.: "Approximate Traveling Salesman Algorithms", Operations Research 28, 694-711 (1980).

[52] GOLDEN, B., MAGNANTI, T.L., NGUYEN, H.Q.: "Implementing Vehicle Routing Algorithms", Networks 7, 113-148 (1977).

[53] GOMORY, R.: "An Algorithm for Integer Solutions to Linear Programs", Recent Advances in Mathematical Programming, 269-302, McGraw-Hill (1963).

[54] GRÖTSCHEL, M.: "On the Symmetric Travelling Salesman Problem: Solution of a 120-City Problem", Mathematical Programming Study 12, 61-77 (1980).

[55] GRÖTSCHEL, M., PADBERG, M.W.: "On the Symmetric Travelling Salesman Problem I: Inequalities", Mathematical Programming 16, 265-280 (1979).

[56] HANDLER, G.Y.: "Minimax Location of a Facility in an Undirected Tree Graph, Transportation Science 7, 287-293 (1973).

[57] HAYS, R.: "The Delivery Problem", Management Science Research Report No. 106, Carnegie Institute of Technology (1967).

[58] HELBIG HANSEN, K., KRARUP, J.: "Improvements of the Held-Karp Algorithm for the Symmetric Travelling Salesman Problem", Mathematical Programming 7, 87-96 (1974).

[59] HELD, M., KARP, R.M.: "A Dynamic Programming Approach to Sequencing Problems", SIAM 10, 196-210 (1962).

[60] HELD, M., KARP, R.M.: "The Travelling Salesman Problem and Minimum Spanning Trees", Operations Research 18, 1138-1162 (1970).

[61] HELD, M., KARP, R.M.: "The Travelling Salesman Problem and Minimum Spanning Trees: Part II"' Mathematical Programming 1, 6-25 (1971).

[62] HELD, M., WOLFE, P., CROWDER, H.D.: "Validation of Subgradient Optimization", Mathematical Programming 6, 62-88 (1974).

[63] HITCHCOCK, F.L.: "The Distribution of a Product from Several Sources to Numerous Localities", Journal of Mathematics and Physics 20, 224-230 (1941).

[64] HONG, S., PADBERG, M.W.: "A note on the Symmetric Multiple Traveling Salesman Problem with Fixed Charges", Operations Research 25, 871-873 (1977).

[65] HOUCK, D., PICARD, J., QUEYRANNE, M., VEMUGANTI, R.: "The Travelling Salesman Problem and Shortest n-Paths", Operations Research 17, 93-109 (1980).

[66] HURTER, A.P. Jr., SCHAEFFER, M.K., WENDELL, R.E.: "Solutions of Constrained Location Problems", Management Science 22, 51-56 (1975).

[67] KARP, R.M.: "Reducibility among Combinatorial Problems", Complexity of Computer Computations, 185-204, Plenum Press (1972).

[68] KRUSKAL, J.B.: "On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem", Proceedings of the American Mathematical Society 2, 48-50 (1956).

[69] LAND, A.H.: "The Solution of Some 100-City Travelling Salesman Problems", article présenté au 10^{e} Symposium international de Programmation Mathématique, Montréal (1979).

[70] LAND, A.H., DOIG, A.G.: "An Automatic Method of Solving Discrete Programming Problems", Econometrica 28, 497-520 (1960).

[71] LAND, A.H., POWELL, S.: "Fortran Codes for Mathematical Programming: Linear, Quadratic and Discrete", Wiley (1973).

[72] LAPORTE, G., NOBERT, Y.: "Le Problème des m Voyageurs de Commerce", Rapport de Recherche no. 78-11, Ecole des Hautes Etudes Commerciales (1978).

[73] LAPORTE, G., NOBERT, Y.: "The Truck Dispatching Problem", article présenté au congrès EURO III, Amsterdam, Hollande (1979).

[74] LAPORTE, G., NOBERT, Y.: "A Cutting Planes Algorithm for the m-Salesmen Problem", Journal of the Operational Research Society 31, 1017-1023 (1980).

[75] LAPORTE, G., NOBERT, Y.: "An Exact Algorithm for Minimizing Routing and Operating Costs in Depot Location", European Journal of Operational Research 6, Special Issue on Locational Decisions, 224-226 (1981).

[76] LAPORTE, G., NOBERT, Y., MERCURE, H.: "The Multi-Depot m-Salesmen Problem", Methods of Operations Research. 367-370, Oelgeschlager, Gunn and Hain Publishers (1981).

[77] LASDON, L.S.: "Optimization Theory for Large Systems", Macmillan (1970).

[78] LAWLER, E.L.: "Branch and Bounds Methods: A Survey", Operations Research 14, 669-719 (1966).

[79] LAWLER, E.L.: "Combinatorial Optimization: Networks and Matroids", Holt, Rinehart et Winston (1976).

[80] LAWRENCE, R.M., PENGILLY, P.J.: "The Number and Location of Depots Required for Handling Products for Distribution to Retail Stores in South-East England", Operational Research Quarterly 20, 23-32 (1969).

[91] LEA, A.C.: "Location-Allocation Systems: An Annotated Bibliography", Discussion Paper No. 13, Department of Geography, University of Toronto (1973).

[82] LEA, A.C.: "A Model Taxonomy and a View of Research Frontiers in Normative Modelling", Paper presented to the International Symposium on Locational Decisions, Banff, Alberta (1978).

[83] LENSTRA, J.K., RINNOY KAN, A.H.G.: "Some Simple Applications of the Travelling Salesman Problem", Operational Research Quarterly 26, 717-733 (1975).

[84] LIN, S.: "Computer Solutions of the Traveling Salesman Problem", Bell System Technical Journal 44, 2245-2269 (1965).

[85] LITTLE, D.C., MURTY, K.G., SWEENEY, D.W., KAREL, C.: "An Algorithm for the Travelling Salesman Problem", Operations Research 11, 972-989 (1963).

[86] MADSEN, O.B.G.: "A Survey of Methods for Solving Combined Location-Routing Problems", Preprints of the Second International Symposium on Locational Decisions, 373-386, Skodsborg, Denmark (1981).

[87] MAGNANTI, T.L.: "Combinatorial Optimization and Vehicle Fleet Planning: Perspectives and Prospects", Network 11, 179-213 (1981).

[88] MARTIN, G.: "An Accelerated Euclidean Algorithm for Integer Linear Programming", Recent Advances in Mathematical Programming, 311-318, McGraw-Hill (1963).

[89] MARTIN, G.: "Solving the Travelling Salesman Problem by Integer Linear Programming", Working Paper, Control Data Corporation (1966).

[90] MILIOTIS, P.: "Combining Cutting Planes and Branch-and-Bound Methods to Solve Integer Programming Problems: Applications to the Travelling Salesman Problem and the 1-Matching Problem", University of London, Ph.D. (1975).

[91] MILIOTIS, P.: "Integer Programming Approaches to the Travelling Salesman Problem", Mathematical Programming 10, 367-378 (1976).

[92] MILIOTIS, P.: "Using Cutting Planes to Solve the Symmetric Travelling Salesman Problem", Mathematical Programming 15, 177-188 (1978).

[93] MILIOTIS, P., LAPORTE, G., NOBERT, Y.: "Computational Comparison of Two Methods for Finding the Shortest Complete Cycle or Circuit in a Graph", RAIRO (recherche opérationnelle) 15, 233-239 (1981).

[94] MILLER, C.E., TUCKER, A.W., ZEMLIN, R.A.: "Integer Programming Formulation of Travelling Salesman Problems", JACM 7, 326-329 (1969).

[95] MITTEN, L.G.: "Branch and Bound Methods: General Formulation and Properties", Operations Research 18, 24-34 (1970).

[96] NORMAN, R.Z.: "On the Convex Polyhedra of the Symmetric Travelling Salesman Problem", Bulletin of the American Mathematical Society 61, 559 (1955).

[97] O'BRIEN, H., Letter to the Editor, Operational Research Quarterly 28 (1977).

[98] PADBERG, M.W., HONG, S.: "On the Symmetric Travelling Salesman Problem: A Computational Study", Mathematical Programming Study 12, 78-107 (1980).

[99] PALMER, D.S.: "The Placing of Service Points to Minimize Travel", Operational Research Quarterly 24, 121-123 (1973).

[100] PIERCE, J.F.: "Direct Search Algorithms for the Truck-Dispatching Problem", Transportation Research 3, 1-42 (1969).

[101] RAND, G.K.: "Methodological Choices in Depot Location Studies", Operational Research Quarterly 27, 241-249 (1976).

[102] RAO, M.R.: "A Note on the Multiple Travelling Salesman Problem", Operations Research 28, 628-632 (1980).

[103] RAO, M.R., ZIONTS, S.: "Allocation of Transportation Units to Alternative Trips - A Column Generation Scheme with Out-of-Kilter Subproblems", Operations Research 16, 52-63 (1968).

[104] RAPP, Y.: "Planning of Exchange Locations and Boundaries", Ericsson Technics 12, 1-22 (1962).

[105] ROSS, G.T., SOLAND, R.M.: "A Branch and Bound Algorithm for the Generalized Assignment Problem", Mathematical Programming 8, 91-103 (1975).

[106] ROSS, G.T., SOLAND, R.M.: "Modeling Facility Location Problems as Generalized Assignment Problems", Management Science 24, 345-357 (1977).

[107] SAHNI, S., GONZALES, T.: "P-Complete Approximation Problems", JACM 23, 555-565 (1976).

[108] SALKIN, H.M.: "Integer Programming", Addison-Wesley (1975).

[109] SANDI, C.: "Subgradient Optimization", Combinatorial Optimization, 73-91, Wiley (1979).

[110] SHAEFFER, M.K., HURTER, A.P. Jr.: "An Algorithm for the Solution of a Location Problem with Metric Constraints", Naval Research Logistic Quarterly 21, 625-636 (1974).

[111] SHAPIRO, D.: "Algorithms for the Solution of the Optimal Cost and Bottleneck Travelling Salesman Problems", Washington University, Sc.D. (1966).

[112] SMALLWOOD, R.D.: "Minimax Detection Station Placement", Operations Research 13, 636-646 (1965).

[113] SMITH, T.H.C.: "A LIFO Implicit Enumeration Algorithm for the Asymmetric Travelling Salesman Problem Using a One-Arborescence Relaxation", Mathematical Programming Study 12, 108-114 (1980).

[114] SMITH, T.H.C., THOMPSON, G.L.: "Computational Performance of Three Subtour Elimination Algorithms for Asymmetric Traveling Salesman Problems", Annals of Discrete Mathematics 1, 495-506 (1977).

[115] STOER, J., WITZGALL, C.: "Convexity and Optimization in Finite Dimensions I", Springer (1970).

[116] SUSSAMS, J.E.: "Efficient Road Transport Scheduling", Gower Press (1971).

[117] SVESTKA, J.A.: "Response to *A Note on the Formulation of the M-Salesman Traveling Salesman Problem*", Management Science 22, 706 (1976).

[118] SVESTKA, J.A.: "A Continuous Variable Representation of the Travelling Salesman Problem", Mathematical Programming 15, 211-213 (1978).

[119] SVESTKA, J.A., HUCKFELDT, V.E.: "Computational Experience with an M-Salesman Traveling Salesman Algorithm", Management Science 19, 790-799 (1973).

[120] TILLMAN, A.F., BRADFORD, H.O.: "An Application of an Optimal Distribution Scheduling System", paper presented at the ORSA Conference on Scheduling, Orlando, Florida (1976).

[121] TILLMAN, F., CAIN, T.: "An Upper Bound Algorithm for the Single and Multiple Terminal Delivery Problem", Management Science 18, 664-682 (1972).

[122] TOMIZAWA, N.: "On Some Techniques Useful for Solution of Transportation Network Problems", Networks 1, 173-194 (1971).

[123] VOLGENANT, T., JONKER, R.: "A Branch and Bound Algorithm for the Symmetric Traveling Salesman Problem based on the 1-Tree Relaxation", European Journal of Operational Research 9, 83-89 (1982).

[124] WEBB, M.H.J.: "Cost Functions in the Location of Depots for the Multiple-Delivery Journeys", Operational Research Quarterly 19, 311-320 (1968).

[125] WEBER, A.: "Alfred Weber's Theory of the Location of Industries (translated from German by C.J. Friedrich)", University of Chicago Press (1929).

[126] WREN, A., HOLLIDAY, A.: "Computer Scheduling of Vehicles from One or More Depots to a Number of Delivery Points", Operational Reserach Quarterly 23, 333-344 (1972).